

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES À FINALITÉ SPÉCIALISÉE EN SOFTWARE ENGINEERING

Comprendre l'ordinateur à l'aide d'un système informatique tangible

le micro:bit, au cours d'une séquence d'activité à destination de jeunes de 12 à 15 ans

THEATE, Nicolas

Award date:
2020

Awarding institution:
Université de Namur

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

UNIVERSITÉ DE NAMUR
Faculté d'informatique
Année académique 2019–2020

**Comprendre l'ordinateur à l'aide d'un
système informatique tangible: le micro:bit,
au cours d'une séquence d'activité à
destination de jeunes de 12 à 15 ans**

Nicolas Théate



Maître de stage : Julie HENRY

Promoteur : _____ (Signature pour approbation du dépôt - REE art. 40)
Benoît FRÉNAY

Co-promoteur : Julie HENRY

Mémoire présenté en vue de l'obtention du grade de
Master en Sciences Informatiques.

Résumé

Dans une société où le numérique se fait de plus en plus présent, il est impensable aujourd'hui pour un jeune de s'en passer tant elle lui facilite le quotidien. Et pourtant, leurs visions de la technologie et de l'informatique restent floues. Il devient donc nécessaire de démystifier l'ordinateur, afin d'aider ceux-ci à appréhender ce monde toujours plus connecté.

La question de recherche présentée dans ce mémoire aborde l'élaboration et l'efficacité d'une séquence d'activité centrée sur l'initiation à l'informatique et à la programmation auprès des élèves issus du secondaire inférieur (12 à 15 ans).

La principale particularité de ces activités est l'utilisation d'un système informatique tangible : le micro:bit, un micro-ordinateur composé de plusieurs capteurs et programmable à l'aide d'une interface web de programmation par blocs.

Mots-clés : Ordinateur - micro:bit - programmation - éducation numérique - système tangible

Abstract

In a society where digital technology is getting more and more present and simplifies our every day life, it is unthinkable for the youth to live without it. And yet, their vision of technology and computing remain blurred. It therefore becomes necessary to demystify computers and help this new generation grasps our increasingly connected world.

The research question presented in this master thesis addresses the development and effectiveness of a sequence of activities aiming on introducing high school students (from age 12 to 15) to computers and programming.

The main particularity of these activities is the use of a tangible computing system : the micro:bit, a microcomputer made of several sensors, and programmable via a block programming web interface.

Keywords : computer - micro:bit - programming - digital education - tangible system

Avant-Propos

Ce mémoire est le résultat d'un travail effectué au cours d'un stage à l'Université de Namur.

Plusieurs activités et tests ont été effectués dans différentes écoles, dont je remercie les directeurs et professeurs d'avoir accepté de nous accueillir, ainsi que les élèves pour leur participation et leurs engagements dans nos activités.

Je tiens à remercier Benoît Frénay, pour sa patience et sa disponibilité, ainsi que Julie Henry, maître de stage et co-promotrice de ce travail.

Je tiens aussi à remercier les membres du jury pour l'intérêt qu'ils porteront à mon travail.

Enfin, je tiens à remercier ma sœur et mes parents de m'avoir soutenu durant ces années d'études, ainsi que mes amis fait en cours de route.

Table des matières

I	Introduction	1
II	Contexte et état de l'art	4
1	L'enseignement de l'informatique	5
1.1	La situation en Belgique	5
1.2	Situation ailleurs	6
2	Solutions Technologiques	7
2.1	Programmation par blocs	7
2.2	Objets tangibles programmables	8
2.2.1	Les robots	8
2.2.2	Les cartes programmables	10
III	Outils et concepts visés par la recherche	11
3	Le cas micro :bit	12
3.1	Qu'est-ce que le micro :bit ?	12
3.2	Tour d'horizon des activités autour du micro :bit	14
3.3	Pourquoi ce choix ?	15
4	Concepts d'informatique	17
4.1	Quels concepts ?	17
4.2	Comment enseigner ces concepts ?	19
IV	Développement et validation des fiches d'activité	21
5	Méthodologie	22
5.1	Cadre de la méthodologie	22
5.2	Méthode d'évaluation	23
6	Fiches d'activité	24
6.1	Présentation de la séquence d'activités	24
6.2	Présentation des fiches	25

6.2.1	Les fiches "prof"	25
6.2.2	Les fiches explicatives	25
6.2.3	Les fiches "élève"	25
6.2.4	Les fiches "aide"	25
6.3	Présentation des activités	26
6.3.1	Activité 1 : "Premiers pas avec le micro :bit – Découverte des entrées et sorties"	26
6.3.2	Activité 2 : "Premiers pas avec le micro :bit – Introduction à la programmation"	28
6.3.3	Activité 3 : "Variable et accéléromètre"	28
6.3.4	Activité 4 : "Conditionnelle et boussole"	29
7	Retours et résultats	31
7.1	Retour des élèves	31
7.2	Analyse des résultats	31
V	Conclusion	37
	Bibliographie	39
A	Fiche "prof" : Premiers pas avec le micro :bit - Découverte des entrées et sorties	40
B	Fiche "prof" : Premiers pas avec le micro :bit - Introduction à la programmation	45
C	Fiche "prof" : Variable et accéléromètre	49
D	Fiche "prof" : Conditionnelle et boussole	53
E	Fiche explicative : Présentation du micro :bit	57
F	Fiche explicative : Introduction à l'application du micro :bit	59
G	Fiche explicative : Les variables	61
H	Fiche explicative : L'accéléromètre du micro :bit	63
I	Fiche explicative : Les conditionnelles	65
J	Fiche explicative : La boussole, le thermomètre et les capteurs de luminosité du micro :bit	67
K	Fiche "élève" : Découvertes des entrées et sorties	69
L	Fiche "élève" : Introduction à la programmation	71
M	Fiche "élève" : Les variables et l'accéléromètre	73
N	Fiche "élève" : Les conditionnelles et les capteurs	75
O	Fiche "aide" : La calculatrice	77

P	Fiche "aide" : Affichage Météo	79
Q	Contribution à Ludovia#CH 2018	82
R	Proposition d'atelier pour Ludovia#CH 2018	87

Première partie

Introduction

Introduction

Les technologies de l'information et de la communication (TIC) font partie intégrante de notre quotidien. Ordinateurs, tablettes, smartphones, objets connectés, etc. En quelques années seulement, ces outils numériques se sont multipliés et sont devenus presque indispensables, tant ils accompagnent et facilitent la vie de tous. C'est notamment le cas pour la génération Z, née dans un monde devenu numérique. On parle alors de *digital natives* [1].

Cependant, les usages intensifs¹ d'appareils numériques par ces derniers ne garantissent pas forcément la maîtrise de ces outils. Le constat sera fait plus loin dans ce travail que leur vision de l'ordinateur est assez réduite lorsque, par exemple, ils ne citent pas leur smartphone comme exemple d'ordinateur. Sachant cela, il convient alors de remettre en question l'usage du terme *digital natives*. «(...) ; au fond, il signifie juste le fait d'avoir grandi dans un environnement numérique. Cet étiquetage écrase toute hétérogénéité, la différence entre ceux qui sont à l'aise et ceux dont les connaissances sont plus fragiles dès qu'ils quittent l'univers de leurs applications.» (Glassey²)

Le nom *digital natives*³ correspond donc davantage à cette nouvelle génération ayant grandi avec la démocratisation d'outils numériques qu'elle ne connaît pas et ne maîtrise pas totalement. L'image qu'elle se fait d'un ordinateur reste encore assez floue, et ça vaut aussi pour son fonctionnement, ses capacités (et limites) ainsi que sur les moyens pour interagir avec lui.

Or, il devient à présent nécessaire de former les jeunes et de les aider à démystifier l'ordinateur afin qu'ils puissent appréhender ce monde de plus en plus numérique et connecté. "(...) il importe peu de comprendre les détails du fonctionnement très complexe d'un processeur ou d'une carte graphique. Il est par contre essentiel de maîtriser les bases de l'algorithmique et de sa mécanique du raisonnement. Il n'est pas non plus nécessaire de savoir programmer (...). Pour des questions de performance, il peut être utile de comprendre où l'information que nous utilisons est stockée, en mémoire, sur disque ou sur le réseau. Surtout, il est indispensable de comprendre le sens de cette information, comment elle est représentée, comment elle est organisée." (Abiteboul⁴).

1. 92% des adolescents de 12-17 ans ont un smartphone - <https://www.bva-group.com/sondages/salaries-mobilite-sondage-bva-salesforce-presse-regionale-2/>, consultée en ligne le 20 août 2020

2. <https://www.letemps.ch/societe/lecole-ligne-revele-difficultes-informatiques-digital-natives>, consultée en ligne le 30 mai 2020

3. https://next.liberation.fr/vous/2010/03/10/les-jeunes-ne-sont-plus-interesses-par-l-outil-ordi_614226, consultée en ligne le 20 juillet 2020

4. Leçon inaugurale du Collège de France, prononcée le jeudi 8 mars 2012. Chaire d'Informatique et sciences numériques - <https://books.openedition.org/cdf/529\#ftn1>, consultée en ligne le 17 juillet 2020

Dans ce contexte, la question de recherche proposée est la suivante : une courte séquence d'activité autour de la découverte de l'ordinateur permet-elle aux élèves de 12 à 15 ans de changer la représentation qu'ils se font de l'ordinateur et de l'informatique en général ?

Pour répondre à cela, ce mémoire fera tout d'abord le point sur la situation actuelle de l'éducation au numérique chez les moins de 18 ans en Belgique, ainsi qu'à l'étranger. Un tour d'horizon des initiatives et outils proposés aux jeunes sera ensuite effectué pour conclure l'état de l'art. En partant de cela, la troisième partie aura pour objectif de présenter les outils et concepts qui guideront la recherche. D'une part, en présentant l'outil choisi pour la séquence d'activités, un système informatique tangible appelé le BBC micro:bit⁵, et, d'autre part, en présentant les concepts de base qui forment l'image de l'informatique que l'on souhaite transmettre aux élèves. La méthodologie sera décrite dans la quatrième partie en présentant en détail la séquence d'activités réalisée et se concluant avec une analyse des résultats obtenus à l'aide de pré- et post-tests. Et enfin, la conclusion fera le bilan sur le travail effectué et apportera une réponse à la question de recherche.

5. <https://makecode.microbit.org/>, consultée en ligne le 20 juillet 2020

Deuxième partie

Contexte et état de l'art

Chapitre 1

L'enseignement de l'informatique

Dans ce chapitre, la situation par rapport à l'éducation au numérique, ici et ailleurs, sera présentée. En Belgique francophone dans un premier temps, puis en communauté flamande, avant d'aborder quelques initiatives mises en place à l'étranger.

1.1 La situation en Belgique

En Wallonie, la formation au numérique est quasiment absente du parcours scolaire. J. Henry et N. Joris mettent en évidence dans deux articles [6, 7] la structure hétérogène des programmes d'enseignement de l'informatique. Selon elles, cette disparité se remarque non seulement d'un réseau d'enseignement à un autre, mais également au sein d'un même réseau. Cette hétérogénéité est renforcée par le fait que souvent, les cours d'informatique sont enseignés en option, ce qui amène, suivant les choix personnels des élèves, à une grande disparité des connaissances informatiques en fin de cycle secondaire. À ceci s'ajoute un manque évident de ressources didactiques pour les enseignants, qui se tournent alors vers des supports non validés par des spécialistes [8].

Cependant, la Fédération Wallonie-Bruxelles souhaite réformer le tronc commun avec le Pacte pour un Enseignement d'Excellence¹. Celui-ci sera mis en oeuvre de façon progressive à partir de la rentrée 2020 pour les maternelles et jusqu'en 2027 pour les secondaires. Il sera construit autour de sept domaines d'apprentissage. Le domaine "Formation mathématique, scientifique et technique" contiendra une approche au numérique axée sur les cinq aires de compétences numériques décrites par la Commission européenne [2] : "information and data literacy", "communication and collaboration", "digital content creation", "safety" et "problem solving".

C'est dans le cadre de ce pacte, ainsi que dans celui de Digital Wallonia² (une stratégie numérique pour la Wallonie), que différentes universités et hautes écoles se sont réunies pour former le groupe de travail *SI*² (Sciences Informatiques dans le Secondaire Inférieur). Ce groupe a pour objectif d'introduire les sciences informatiques dans le premier degré du secondaire, en passant notamment par la proposition d'un référentiel de compétences articulé autour de cinq axes : la représentation des données, l'algorithmique, la programmation, le matériel et enfin, les réseaux et la sécurité.

1. <http://www.enseignement.be/index.php?page=28280>, consultée en ligne le 20 août 2020.

2. <https://www.digitalwallonia.be/fr>, consultée en ligne le 20 août 2020.

De l'autre côté de la frontière linguistique, en communauté flamande, les élèves sont mis en contact avec le numérique dès la maternelle [11]. Tandis qu'en primaire et secondaire, le gouvernement flamand a mis en place depuis 2012 le plan d'action STEM³ dans le but d'encourager les jeunes à s'orienter vers des carrières scientifiques comme l'ingénierie et l'informatique.

1.2 Situation ailleurs

Dans un article de 2016 [5], F. Heintz, L. Mannila et T. Färnqvist ont répertorié l'introduction de l'informatique dans l'enseignement secondaire de divers pays. Par exemple, au Royaume-Uni, un nouveau programme⁴ de cours rend l'apprentissage de l'informatique obligatoire dans les écoles depuis septembre 2014. Celui-ci, décliné en trois branches : l'informatique, les technologies de l'information et la culture digitale, présente la programmation comme étant une discipline à part entière. Depuis 2008, on retrouvait déjà une association, Computing at School⁵, dont la vocation était de soutenir l'enseignement de l'informatique et de promouvoir la pensée informatique chez les jeunes élèves en publiant en ligne des ressources.

En Pologne, l'apprentissage de l'informatique fait partie de l'enseignement primaire depuis la fin des années 1960. Au cours des trois premières années, il se mélange à d'autres activités telles que la lecture, le calcul ou le jeu. Pour les trois années suivantes, les outils informatiques sont utilisés comme support aux activités d'études d'autres matières. Au cours des trois dernières années du secondaire, l'apprentissage se concentre sur les algorithmes et la solution de problèmes à l'aide de l'ordinateur. Les étudiants ont le choix entre deux heures/semaine pendant un an ou une heure/semaine pendant deux ans. Ce programme se poursuit durant les trois dernières années de façon plus rigoureuse, avec la possibilité d'un apprentissage poussé de trois heures/semaine avec éventuellement un examen final en informatique en fin de cycle.

Enfin, la Nouvelle-Zélande a vu apparaître une méthode d'enseignement de l'informatique sans ordinateur. Cette méthode, créée par Tim Bell, Ian H. Witten et Mike Fellows, CS Unplugged⁶ est basée sur un programme d'enseignement orienté autour d'activités transmettant des notions de base de l'informatique de façon ludique et sans utiliser d'ordinateurs. Ces activités tournent ainsi autour de thèmes comme la représentation des données en binaire, les algorithmes de recherche et de tri. En proposant différents modes d'enseignement, ce programme dénote d'une réelle envie de toucher le plus grand monde.

3. <https://www.onderwijskieser.be/v2/extra/stem.php>, consultée en ligne le 20 août 2020.

4. <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study>, consultée en ligne le 20 août 2020.

5. <https://www.computingatschool.org.uk/>, consultée en ligne le 20 août 2020.

6. <https://csunplugged.org/en/>, consultée en ligne le 20 août 2020.

Chapitre 2

Solutions Technologiques

Au-delà de l'enseignement, il y a une volonté de créer des initiatives qui éduqueront les jeunes au numérique. Que ce soit de la part de créateurs indépendants proposant entre autres des vidéos de vulgarisation scientifique sur l'informatique ou que ce soit de la part d'entreprises souhaitant mettre à disposition des outils pédagogiques afin de favoriser l'apprentissage de l'informatique. Ce sont quelques-uns de ces outils qui seront présentés dans ce chapitre.

2.1 Programmation par blocs

Un logiciel de programmation par blocs est une interface de programmation présentant les instructions sous la forme de blocs à assembler. Les blocs sont triés par catégorie/par famille. Il suffit alors de les glisser et de les assembler dans la zone de travail pour concevoir aisément un algorithme. Tels des pièces de puzzle, les blocs ont différentes formes et couleurs selon l'instruction qu'ils représentent. Ces formes représentent ainsi les règles syntaxiques, comme des règles d'emboîtement de puzzle, guidant ainsi l'utilisateur et rendant intuitif la combinaison d'instructions.

L'exemple le plus connu de logiciel de programmation par bloc est Scratch¹, un logiciel gratuit développé par le groupe Lifelong Kindergarten du MIT). Spécialement développé pour les enfants de 8 à 16 ans, Scratch permet d'accompagner les jeunes dans la création de jeux, d'animation et d'histoire interactive, poussant ainsi à la créativité et au partage de créations. Son interface simple et intuitive (cfr. Figure 2.1) est gratuite et accessible directement sur la plupart des navigateurs d'ordinateur de bureau ou de tablette. Il existe également une version téléchargeable pour son utilisation hors ligne. Cette interface est découpée en 3 zones : une zone regroupant les blocs par catégorie, une zone de développement dans laquelle l'utilisateur assemblera les blocs pour créer son programme et une zone d'exécution.

1. <https://scratch.mit.edu/>, consultée en ligne le 20 août 2020.

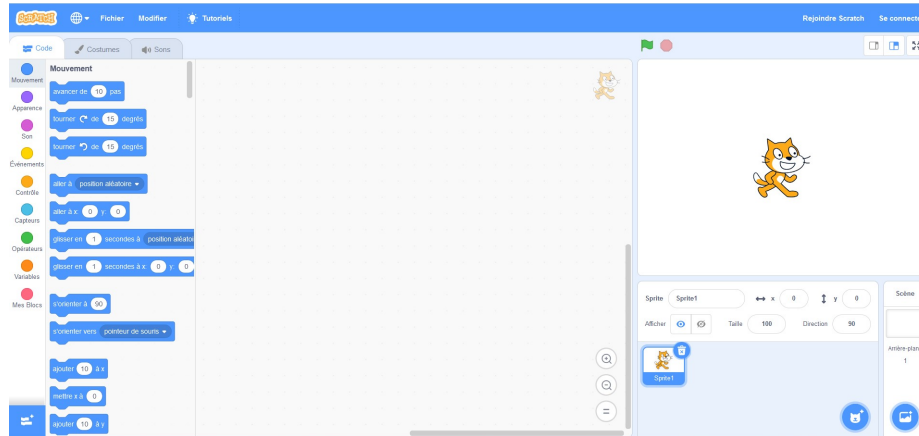


FIGURE 2.1 – L’interface de programmation de Scratch

Lancé en 2006, le logiciel est continuellement mis à jour avec de nouveaux blocs ou l’ajout de nouvelles langues. Plus de 70 langues sont aujourd’hui disponibles. Le site propose également des ressources à destination des jeunes étudiants ou des éducateurs. Ce langage de programmation visuel à base de blocs a déjà démontré son efficacité pour initier de jeunes élèves à l’informatique au cours de précédentes études [9, 12]

2.2 Objets tangibles programmables

Contrairement aux interfaces de programmation, les objets tangibles programmables peuvent être physiquement manipulés par l’utilisateur. Ils permettent notamment de réaliser avec les élèves des activités débranchées, c’est-à-dire sans ordinateur. Deux types d’outils rentrent dans cette catégorie : les robots et les cartes électroniques.

2.2.1 Les robots

Un robot est une machine programmable combinant des composants mécaniques et électroniques. Il a la capacité de percevoir son environnement grâce à des capteurs (observe un signe ou objet, touche quelque chose, capte un changement de température, luminosité, etc.) et de réagir grâce à des actionneurs.

De nombreux mini-robots ont été conçus dans le but d’aider les jeunes à comprendre le numérique comme par exemple, les LEGO Mindstorms², la gamme de robot Dash and Dot³ ou encore le Thymio II⁴.

Thymio est un robot open-source à l’esthétique très neutre né de la collaboration de deux écoles : l’Ecole polytechnique fédérale de Lausanne (EPFL) et l’Ecole cantonale d’art de Lausanne (ECAL) en

2. <https://www.lego.com/fr-fr/themes/mindstorms>, consultée en ligne le 20 août 2020.

3. <https://www.makewonder.com/robots/dash/>, consultée en ligne le 20 août 2020.

4. <https://www.thymio.org/fr/>, consultée en ligne le 20 août 2020.

2011. Il est notamment utilisé dans le cadre du projet Digital Wallonia⁵. Il est muni de multiples capteurs et actionneurs et son prix s'élève à une centaine d'euros.

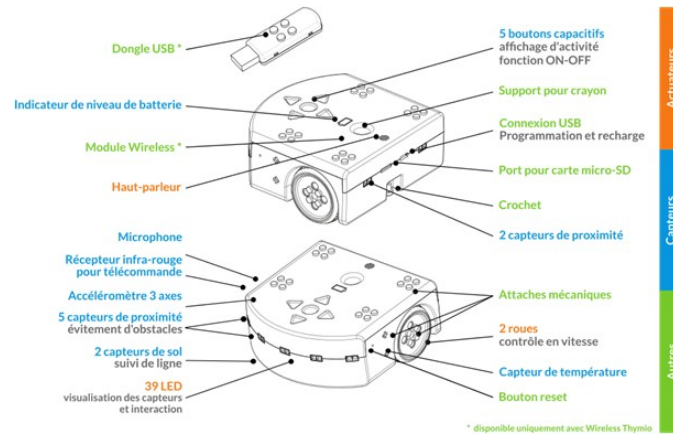


FIGURE 2.2 – Composition du Wireless Thymio
(Source : <http://wiki.thymio.org/fr:thymiospecifications>)

. Thymio est programmable grâce au logiciel Aseba⁶ téléchargeable gratuitement en ligne. Le logiciel permet de programmer le robot en programmation textuelle (en langage Aseba), par blocs (via Blockly ou Scratch) ou visuelle, au moyen du langage VPL⁷. Ce dernier permet d'associer des actions à des événements représentés sous forme de blocs.

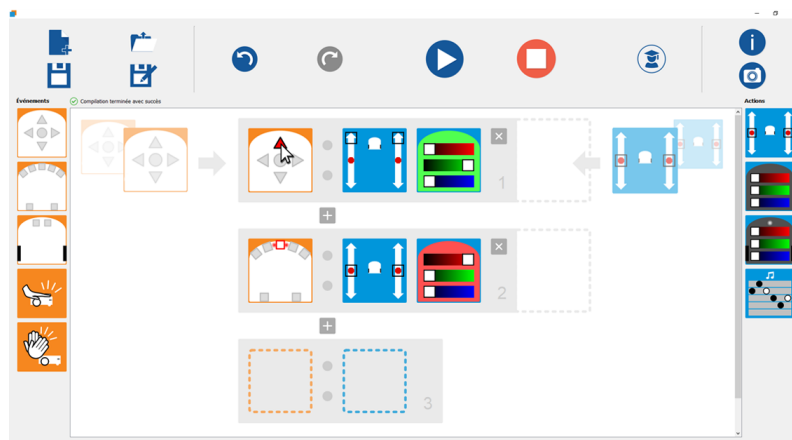


FIGURE 2.3 – Interface de programmation visuelle
(Source : <https://www.thymio.org/fr/programmer/vpl/>)

5. <https://www.digitalwallonia.be/fr/agenda/thymio>, consultée en ligne le 20 août 2020.

6. <https://www.thymio.org/fr/programmer/>, consultée en ligne le 20 août 2020.

7. <https://www.thymio.org/fr/programmer/vpl/>, consultée en ligne le 20 août 2020.

2.2.2 Les cartes programmables

Une carte programmable est une carte électronique sur laquelle se retrouvent des composants dont un ou plusieurs micro-contrôleurs. Pour fonctionner, elle a besoin d'un programme transféré dans sa mémoire à partir d'un ordinateur. Ce programme peut être écrit dans divers langages de programmation.

La carte Arduino⁸ est l'une des cartes les plus connues. Développée à l'Interaction Design Institute d'Ivrea en Italie, cette carte open source et peu coûteuse, conçue pour l'enseignement de l'électronique et de l'informatique, dispose de nombreux connecteurs permettant le branchement de périphériques supplémentaires comme des LEDs, des moteurs, des capteurs, etc.



FIGURE 2.4 – Carte électronique Arduino
(Source : <https://www.arduino.cc/>)

. Sa programmation s'effectue via un logiciel de type IDE fourni par Arduino et basé sur un langage proche du C++. Le transfert du programme sur la carte s'effectue via une connexion USB. Arduino fournit également des packs⁹ de modules prêts à l'emploi comprenant tout le matériel et les programmes nécessaires pour l'apprentissage.

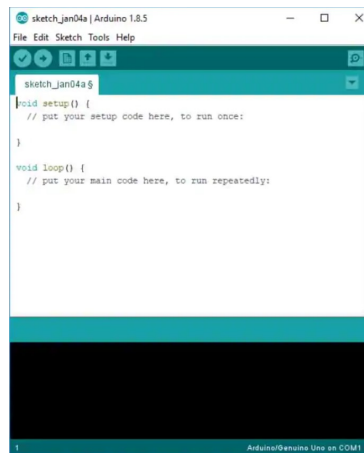


FIGURE 2.5 – Interface de programmation Arduino IDE
(Source : <https://www.arduino.cc/>)

8. <https://www.arduino.cc/>, consultée en ligne le 20 août 2020.

9. <https://www.arduino.cc/en/Main/Products>, consultée en ligne le 20 août 2020.

Troisième partie

Outils et concepts visés par la
recherche

Chapitre 3

Le cas micro:bit

Dans le chapitre précédent, les avantages de l'utilisation de logiciels de programmation par blocs et d'objets tangibles ont été présentés. Ce chapitre va se concentrer sur l'outil micro:bit, comprenant le composant physique ainsi que son interface de programmation.

3.1 Qu'est-ce que le micro:bit ?

Historique

Le micro:bit est un micro-ordinateur développé au Royaume-Uni par la British Broadcasting Corporation (BBC). S'inscrivant dans la lignée du micro-ordinateur Micro de la BBC lancé 30 ans auparavant et inspiré du Raspberry Pi, il fut lancé en juillet 2015 dans le cadre d'une initiative, "BBC's make it digital". Ses objectifs étaient de sensibiliser les jeunes enfants à l'informatique et de les initier à la programmation et ainsi d'aider au développement de futurs informaticiens/"digital pioneers". Au total, ce sont près de 29 partenaires qui soutiennent la BBC pour le développement du micro-ordinateur¹, que ce soit par le développement et la fabrication du micro-ordinateur ou par l'éducation des enfants et la formation des enseignants. Produit pour un faible coût, un million d'exemplaires furent distribués aux enfants de 11-12 ans du Royaume-Uni². Il est également possible au grand public de s'en procurer.

Avec cette initiative, le Royaume-Uni devient le premier pays dans le monde à imposer la programmation informatique dans les établissements primaires et secondaires. En octobre 2016, la fondation à but non lucratif Micro:bit Educational Foundation s'inscrit dans la continuité avec pour objectif d'étendre l'initiative en Europe, puis à l'international. Son objectif en 2020 est de distribuer un micro:bit à près de deux millions d'étudiants des États-Unis et du Canada.

Description de l'appareil

De la taille d'une carte de crédit (51 x 42 x 10 mm), le micro:bit possède une unité centrale composée d'un processeur ARM et d'une mémoire limitée mais suffisante pour contenir un court programme (moins de 300 lignes). Divers périphériques composent ce micro-ordinateur :

- Un écran de 5x5 LEDS rouges.

1. Liste des 29 partenaires - <https://www.bbc.co.uk/mediacentre/mediapacks/microbit/partners>, consultée le 20 août 2020.

2. BBC gives children mini-computers in Make it Digital scheme - <https://www.bbc.com/news/technology-31834927>, consultée le 20 août 2020.

- Deux boutons poussoirs : 'A' et 'B'.
- Un accéléromètre détectant les mouvements.
- Une boussole (ou magnétomètre) détectant les champs magnétiques.
- Un thermomètre.
- Des capteurs de luminosité se situant sous l'écran de LEDs.
- Une antenne Bluetooth
- Une série de broche ('edge connectors') permettant de brancher, à l'aide de pinces crocodile, des périphériques supplémentaires comme un buzzer ou un servomoteur.

Contrairement à certains systèmes "robots" comme le Dash ou le Thymio, les composants du micro:bit sont directement visibles. On retrouve aussi un connecteur micro-USB pour brancher le micro:bit à un ordinateur, un bouton de réinitialisation pour relancer le programme contenu dans la mémoire et une prise pour y brancher une batterie (2 x piles AAA 1,5V) afin de lancer le programme sans être connecté à un ordinateur.

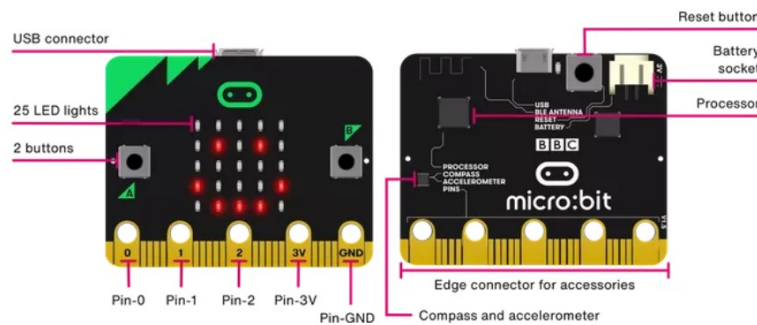


FIGURE 3.1 – Le micro:bit et ses composants, vu de face et de dos.
(Source : <https://microbit.org/get-started/user-guide/overview/>)

Le micro:bit permet l'exécution d'un programme écrit en langage Javascript, Python, ou réalisé à l'aide d'un éditeur développé par Microsoft : MakeCode³. Ce dernier est une interface web de programmation par blocs, accessible gratuitement en ligne et traduite dans une trentaine de langues dont le français.

Avant de compiler son programme, l'utilisateur peut le tester instantanément sur le micro:bit virtuel, simulateur représentant chaque aspect du micro-ordinateur, présent dans l'interface. Une fois satisfait de son programme, il peut le télécharger (fichier 'microbit-<nom du programme>.hex') et le transférer manuellement sur le micro:bit (connecté à l'ordinateur par un câble micro USB). Le programme se chargera et se lancera automatiquement dès la mise sous tension de l'appareil. Le micro:bit ne pouvant contenir qu'un programme à la fois, chaque nouveau programme copié sur l'appareil effacera automatiquement le précédent.

L'interface de programmation est aussi disponible sous forme d'exécutable. Celui-ci présente deux avantages par rapport à sa version en ligne : une connexion internet n'est plus nécessaire pour profiter de l'interface et le transfert du programme vers le micro:bit est grandement facilité. Il suffit en effet de cliquer sur le bouton 'Télécharger' pour transférer directement le programme sur le micro:bit branché, facilitant ainsi les manipulations.

3. <https://makecode.microbit.org/>, consultée le 20 août 2020.

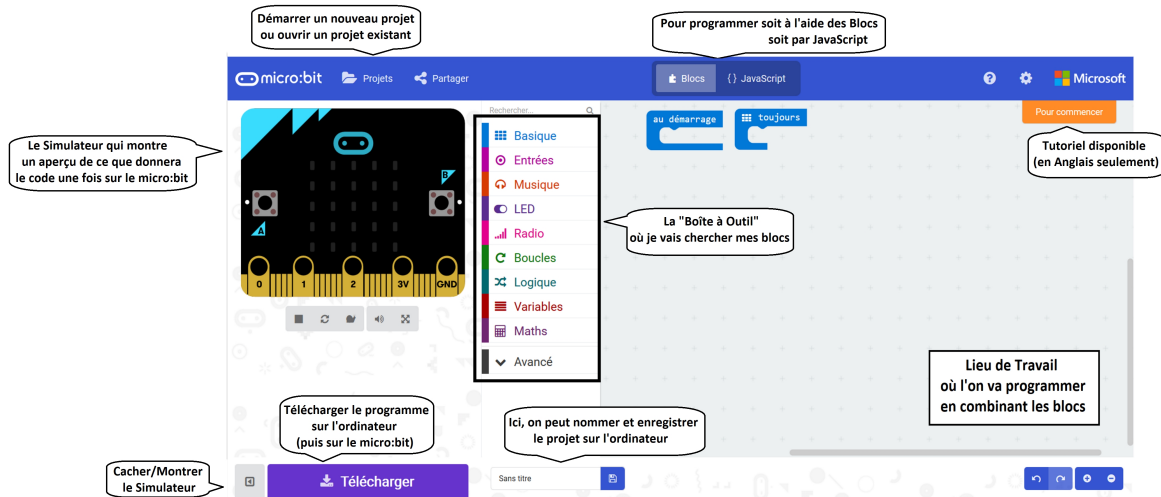


FIGURE 3.2 – Interface de programmation par bloc

Cette interface permet de passer de la programmation par blocs à un éditeur en Javascript ou Python. Cette fonctionnalité permet de se familiariser avec les instructions et la structure plus évoluées qu'offrent ces langages.

D'autres éditeurs de code en ligne permettent aussi de créer des programmes pour le micro:bit. Edublocks⁴ est assez similaire à MakeCode. Les éditeurs MicroPython⁵ et Mu⁶ s'orientent, eux, davantage vers la programmation de code en Python à destination du micro:bit. L'éditeur MicroPython est disponible en ligne uniquement tandis que Mu doit être téléchargé et installé en local. L'IDE Arduino permet aux habitués de créer des programmes à destination du micro-ordinateur. Scratch⁷ est lui aussi compatible avec le micro:bit permettant la création de jeux exploitant ses fonctionnalités.

Des applications Android et iOS sont également disponibles. Au moyen d'un appareil mobile avec une connexion internet et compatible 'Bluetooth Smart Technology', l'application permet de réaliser un code, de le compiler et de l'envoyer au micro:bit par connexion Bluetooth.

3.2 Tour d'horizon des activités autour du micro:bit

Pour accompagner le micro:bit, la BBC propose plusieurs ressources⁸ (documentation, ressource, etc.) afin d'accompagner les enfants dans la découverte et l'apprentissage de l'appareil et de l'interface de programmation. Les capacités du micro-ordinateur ont mené à de nombreux projets éducatifs autour du micro:bit.

Le projet pierre-papier-ciseaux⁹ en est un exemple populaire. Il consiste à réaliser un programme affichant aléatoirement une pierre, des ciseaux ou une feuille en secouant l'appareil. Ces projets sont simples à réaliser et permettent d'accompagner les enfants dans leurs premiers pas de la programmation

4. <https://app.edublocks.org/#MicroBit>, consultée le 20 août 2020.

5. <https://python.microbit.org/v/2.0>, consultée le 20 août 2020.

6. <https://codewith.mu/>, consultée le 20 août 2020.

7. <https://scratch.mit.edu/microbit>, consultée le 20 août 2020.

8. <https://makecode.microbit.org/docs>, consultée le 20 août 2020.

9. <https://makecode.microbit.org/projects/rock-paper-scissors>, consultée le 20 août 2020.

de l'appareil. Certaines séries télévisées de la BBC ont même été mises à contribution afin de populariser l'initiative au travers d'une série d'activités interactives. Ainsi, dans la série de science-fiction 'Doctor Who' le micro-ordinateur a aidé le Docteur à protéger la terre dans "Doctor Who and the micro:bit Live Lesson" ¹⁰.

Parmi les activités les plus populaires, on retrouve des activités exploitant l'aspect tangible du micro:bit pour ajouter une part de créativité aux projets en incluant, par exemple, le micro:bit dans des bricolages faits maison. Ces activités utilisent notamment des pinces crocodiles connectées aux broches P0, P1, P2 et GND du micro:bit et de l'aluminium comme par exemple, la création d'un robot en carton (Figure 3.4) réagissant à la lumière en utilisant un servomoteur ¹¹ ou d'instrument de musique en connectant un buzzer au micro:bit.

On peut ainsi combiner la programmation avec d'autres sciences comme la biologie la création d'un capteur d'humidité du sol ¹² en utilisant des pinces crocodiles et des clous (Figure 3.4).

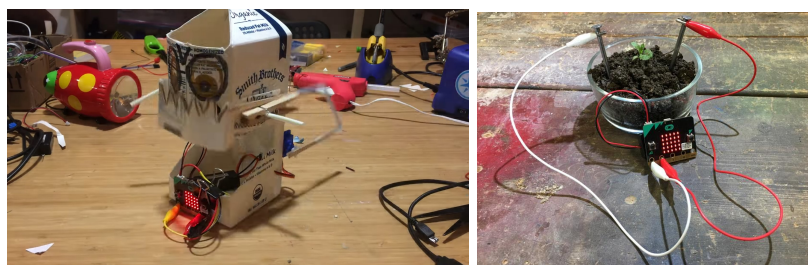


FIGURE 3.3 – Exemples de 'bricolages' avec le micro:bit
(Source : <https://makecode.microbit.org/projects/>)

Ces activités sont de bons exemples d'activités permettant de montrer les multiples formes que peuvent prendre un ordinateur, au-delà de celle d'un ordinateur de bureau.

Enfin, afin de mettre en avant le côté inter-connecté d'un ordinateur, des activités basées sur les fonctionnalités radio du micro:bit ¹³ ont été développées, notamment, envoyer un message depuis un micro:bit à d'autres se trouvant à proximité.

3.3 Pourquoi ce choix ?

Avec son petit prix et son interface web de programmation gratuite, attrayante et intuitive, le micro:bit se présente comme un outil intéressant pour le développement d'activités centrées sur la découverte de l'ordinateur. Le micro-ordinateur a déjà conquis de nombreux élèves et enseignants. Un an après son lancement, une étude réalisée au Royaume-Uni auprès de 405 élèves a révélé que, pour la majorité d'entre eux, le micro:bit les avait aidés à avoir un nouveau regard sur la programmation ¹⁴. L'un des aspects du micro:bit qui justifie son choix est la volonté de mettre à disposition des enfants un système tangible qui leur permette de faire facilement le lien entre les différents composants de ce dernier et le côté abstrait de la programmation : « Even something as simple as projecting a display on

10. <https://www.bbc.co.uk/programmes/articles/3ydvd6mvhl89cHVJ7F2nmzf/doctor-who-and-the-micro-bit-live-lesson>, consultée le 20 août 2020.

11. <https://makecode.microbit.org/projects/milky-monster>, consultée le 20 août 2020.

12. <https://makecode.microbit.org/projects/soil-moisture>, consultée le 20 août 2020.

13. <https://makecode.microbit.org/projects/radio-games>, consultée le 20 août 2020.

14. <https://www.bbc.co.uk/mediacentre/latestnews/2017/microbit-first-year>, consultée le 20 août 2020.

top of paper, or on a tabletop with physical objects, could be used as an effective technique for getting children to see the relationships between objects and representations they create and manipulate in the physical world, and what happens in the computational world. » [10]. L'intérêt du micro:bit est sans conteste grandissant : « the results from this research assert that pupils believe that the BBC micro :bit is an enjoyable, easy to use device that is beneficial when learning how to code, with various other skills being developed in tandem » [3].

Chapitre 4

Concepts d'informatique

Ce chapitre a pour but de répondre aux questions suivantes : quels concepts souhaitons-nous transmettre aux élèves ? Et par quels moyens leur présenter ?

4.1 Quels concepts ?

Dans le référentiel de compétence élaboré par le groupe *SI²*, cinq axes de compétences sont décrits comme essentiels à l'apprentissage du fonctionnement de l'ordinateur et de ses capacités :

- la représentation des données : comprendre comment les informations (nombres, caractères, etc.) sont représentées pour un ordinateur ne fonctionnant qu'à partir de 0 et de 1.
- L'algorithmique : comprendre que toute résolution d'un problème peut être décomposée en étapes.
- La programmation : comprendre la traduction d'un algorithme en langage de programmation compris par l'ordinateur.
- Le matériel : comprendre les composants de base de la machine pour appréhender les éléments essentiels de son fonctionnement
- Les réseaux et la sécurité : comprendre dans les grandes lignes le fonctionnement des réseaux

Parmi ces axes, le matériel, l'algorithmique et la programmation seront mis en avant par la séquence d'activités composées dans le cadre de ce mémoire.

L'objectif n'est pas ici de donner toutes les bases de l'informatique mais de se concentrer surtout sur la question "Qu'est-ce qu'un ordinateur ?" afin de démystifier l'image que les jeunes en font. Les élèves devront être capables de lister les composants de base de l'ordinateur et de comprendre leurs fonctions et leurs rôles dans l'exécution d'un programme. Le choix de la représentation de l'ordinateur que l'on souhaite transmettre s'est porté sur celle d'une machine (électronique) capable d'exécuter une suite d'actions pour lesquelles elle a été programmée. Cette machine peut être décomposée en trois parties : l'unité centrale, composée du processeur et de la mémoire qui interagissent entre eux, les périphériques d'entrée et les périphériques de sortie qui rendent possible les interactions entre la machine et l'utilisateur. Les composants de cette machine sont clairement identifiables sur son schéma fonctionnel (cfr. Figure 4.1).

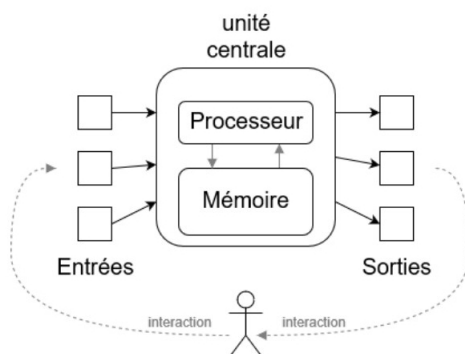


FIGURE 4.1 – Schéma fonctionnel de l'ordinateur

Il est important de mettre en avant les concepts d'entrée et de sortie d'un ordinateur car c'est par ses périphériques que nous, utilisateurs, interagissons avec lui. En comprenant bien cela, les jeunes élèves ont une meilleure compréhension de leur environnement et des technologies qu'ils utilisent.

Après avoir montré ce qu'est un ordinateur, les axes de l'algorithmique et de la programmation seront abordés par la séquence d'activités. Ces deux axes sont intimement liés puisque l'on peut représenter la programmation comme le fait de traduire un algorithme en programme pouvant être lu et exécuté par la machine. Le concept de programmation peut ainsi être simplifié sous la représentation d'une liste d'instructions que l'ordinateur exécute pour atteindre un objectif. Parmi les nombreuses instructions existantes utilisées en programmation, le choix a été fait de présenter aux élèves les concepts de variable et de conditionnelle.

On définit alors une variable comme étant une case dans la mémoire de l'ordinateur portant un nom et pouvant contenir une valeur. Elle peut stocker un nombre mais aussi des symboles comme des lettres ou des messages entiers sous forme de suite de caractères. Son contenu dépend du type de la variable. On peut donc mettre une valeur dans une variable et en modifier son contenu mais aussi récupérer la valeur qui se trouve dans la variable.

Le concept de conditionnelle est assimilable à une instruction où une condition est définie et a pour valeur 'vrai' ou 'faux'. Si cette condition est 'vraie', alors l'instruction va exécuter une "action 1" mais si la condition est 'fausse', alors l'instruction va exécuter une "action 2". En fonction de la condition, le programme exécutera soit l'"action 1", soit l'"action 2".

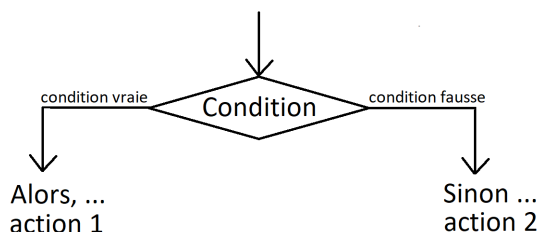


FIGURE 4.2 – Schématisation du concept de conditionnelle

4.2 Comment enseigner ces concepts ?

Afin d'enseigner ces concepts, l'utilisation d'analogie se révèle comme une évidence. Les analogies comparent les similarités entre un concept familier et un concept inconnu. Glynn rappelait encore en 2008 l'importance pour les professeurs de sciences d'utiliser des analogies comme un outil pédagogique pour expliquer des concepts scientifiques importants [4].

Ainsi, pour expliquer de façon imagée le schéma fonctionnel de l'ordinateur, une analogie avec le corps humain est choisie. La carte mère étant le squelette, la structure. Le processeur est le cerveau, traitant les informations qu'il reçoit et interagissant avec la mémoire. Les périphériques d'entrées du corps humain étant nos cinq sens nous permettant de recueillir des informations de notre environnement (oeil=webcam, micro=oreilles, etc.) qui seront traitées par notre cerveau, le processeur du corps humain. Ce dernier interagit avec la mémoire et traite les informations qu'il reçoit pour en sortir de nouvelles informations/instructions dirigées vers nos périphériques de sortie : les muscles, les cordes vocales, etc.

Si le schéma fonctionnel permet aux élèves d'avoir une idée globale de ce qu'est un ordinateur, il est intéressant de le mettre en parallèle avec ce que les élèves connaissent de leur environnement, notamment en leur demandant si, selon eux, un appareil (smartphone, objets connectés, etc.) est un ordinateur ou encore en de citer des exemples qu'ils connaissent d'entrées et de sorties existantes.

Le concept de programmation peut être mis en parallèle avec une liste d'instructions à suivre pour réaliser un avion à partir d'une feuille de papier. Une des analogies les plus connues de la programmation est celle de la recette de cuisine. Dans cette analogie, les ingrédients de la recette sont les paramètres initiaux du programme. Les instructions de la recette sont les étapes à suivre dans un ordre précis pour obtenir, après exécution, un plat cuisiné à partir des ingrédients.

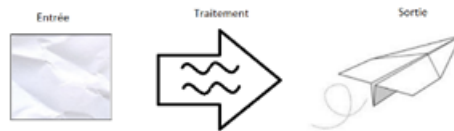


FIGURE 4.3 – Schématisation du concept de programmation

Pour faire comprendre ce qu'est une variable et son fonctionnement, les élèves peuvent réaliser un exercice sur papier, à savoir une suite de différentes instructions. Ils exécutent alors les instructions comme le ferait un ordinateur, de façon séquentielle, créant et modifiant des variables et en allant chercher l'information nécessaire au bon endroit.

Étape	Instruction	Mémoire
1	a = 5;	a: 5
2	b = 2;	a: 5 b: 2
3	a = a + 3;	a: 8 b: 2
4	c = a * b;	a: 8 b: 2 c: 16
5	a = c;	a: 16 b: 2 c: 16
6	d = "Val";	a: 16 b: 2 c: 16 d: "Val"

FIGURE 4.4 – Schématisation du concept de variable

Enfin, pour le concept de conditionnelle, il suffit simplement de le mettre en parallèle avec une situation de la vie commune telle que "SI il pleut, ALORS prends ton parapluie SINON tu peux aller jouer au tennis."

Dans le chapitre 6, la séquence d'activités conçue à partir de l'outil micro:bit et des concepts présentés dans ce chapitre sera présenté en détail.

Quatrième partie

Développement et validation des fiches d'activité

Chapitre 5

Méthodologie

Ce chapitre aborde la méthodologie employée pour évaluer l'efficacité d'une séquence d'activités exploitant le micro:bit. Cela s'est fait dans le cadre du projet «School-IT»¹, un projet initié par la faculté d'informatique de l'UNamur, l'université de Namur, ayant pour but de développer plusieurs ressources (fiches pédagogiques, etc.) à destination des enseignants qui souhaitent enseigner de l'informatique à de jeunes élèves.

5.1 Cadre de la méthodologie

Afin de vérifier si des activités orientées utilisant le micro:bit pour mettre en avant certains concepts permettent de changer le regard qu'ont les jeunes de l'ordinateur, une séquence d'activités a été développée afin de transmettre au mieux la représentation simplifiée de l'ordinateur décrites dans le chapitre 4. Cela, en mettant en jeu les différents périphériques du micro:bit, notamment ses capteurs, présentés dans le chapitre 3. Cette séquence est composée de quatre activités. Chacune de ces activités est déclinée sous forme de fiche pédagogique. Ces activités et fiches sont présentées en détail dans le chapitre 6.

La séquence d'activités a été réalisée auprès de 200 élèves de première et deuxième secondaire (12 - 15 ans) de l'enseignement belge. Parmi les classes ayant participé aux activités, on regroupe :

- Trois classes de 1ère de l'Institut Technique de Namur dont deux classes en enseignement différencié.
- Deux classes de 1ère du Centre scolaire Sainte-Marie de Namur.
- Quatre classes de 2ème de l'Institut de la Providence de Champion.
- Deux classes de 2ème de l'Athénée Royal de Namur.
- Une classe de 1ère de l'Institut Saint-Berthuin Malonne.

La séquence a été réalisée de façon à ce que chaque activité dure une heure de cours et s'est déroulée à raison d'une activité par semaine, à l'exception des élèves de première différenciée qui ont réalisé deux activités par semaine, lors de sessions de deux heures. Dans le cadre de cette recherche, les activités ont été données par des informaticiens sans formation en pédagogie mais ont été conçues pour qu'elles puissent être données par des enseignants n'ayant pas de formation en informatique.

1. <https://school-it.info.unamur.be>, consultée en ligne le 20 août 2020.

5.2 Méthode d'évaluation

Pour récolter des données, un questionnaire a été élaboré sur base du modèle pré-test et post-test². Avant chaque début de séquence d'activités, les élèves répondent à un pré-test afin d'évaluer leurs connaissances et leur vision de l'ordinateur et des métiers informatiques. Au bout des quatre activités de la séquence, les élèves répondent à un post-test, similaire en tout point au pré-test afin de mesurer l'évolution de leur vision de la machine. Ces pré- et posts-tests sont composés de deux parties.

D'une part, des questions ouvertes pour percevoir l'image que les élèves se font de l'informatique et des métiers informatiques en général ainsi que de l'ordinateur et de ses limites.

- "Pour toi, l'informatique, c'est ..."
- "A quoi penses-tu quand tu entends 'métiers informatiques'?"
- "Selon toi, quels sont les points positifs et négatifs dans les 'métiers informatiques'?"
- "Pour toi, un ordinateur, c'est ..."
- "Selon toi, qu'est-ce qu'un ordinateur peut/ne peut pas faire?"

D'autre part, un vrai ou faux sur certains clichés autour des métiers informatiques afin déterminer la représentation que les élèves s'en font et évaluer l'impact de la séquence d'activités sur ceux-ci :

- "Travailler dans l'informatique, c'est toujours faire la même chose"
- "Travailler dans l'informatique, c'est travailler derrière un PC"
- "Travailler dans l'informatique, c'est uniquement taper des lignes de code"
- "Les gens qui travaillent dans l'informatique n'ont pas de vie sociale."
- "Pour travailler dans l'informatique, il faut tout savoir sur les ordinateurs."
- "Pour travailler dans l'informatique, il faut être surdoué(e) en maths."
- "L'informatique, c'est pour les hommes."

Une analyse des résultats obtenus sera menée au chapitre 7, en comparant les résultats des pré et post-tests et au moyen du logiciel IraMuTeQ³ permettant de tirer des conclusions quant à l'efficacité de la séquence d'activités sur les élèves.

2. <https://explorable.com/pretest-posttest-designs>, consultée en ligne le 20 août 2020.

3. <http://iramuteq.org/>, consultée en ligne le 20 août 2020.

Chapitre 6

Fiches d'activité

Dans ce chapitre, la séquence d'activités réalisée avec les élèves et décrite sous forme de fiches d'activité sera présentée tant dans la forme que dans le fond.

6.1 Présentation de la séquence d'activités

Quatre activités ont été développées afin de composer la séquence. Celles-ci mettent notamment en avant les différents périphériques d'entrée et de sortie du micro:bit ainsi que les concepts d'informatique que l'on souhaite transmettre comme décrit au chapitre 5. L'ordre des activités de la séquence est à respecter bien que les activités 3 et 4 soient interchangeables.

Intitulé	Description	Objectifs
Premiers pas avec le micro:bit - Découverte des entrées et sorties	Découvrir les capacités du micro:bit à partir de programmes préinstallés et identifier les entrées et sorties	Apprendre à manipuler le micro:bit et savoir identifier ses composants
Premiers pas avec le micro:bit - Introduction à la programmation	Découvrir l'interface web de programmation par blocs associée au micro:bit	Savoir programmer à l'aide de l'interface web et transférer un programme sur le micro:bit
Variable et accéléromètre	Réaliser une calculatrice utilisant variables et capteur (ici, l'accéléromètre)	Comprendre et savoir utiliser les variables et l'accéléromètre
Conditionnelle et boussole	Réaliser à l'aide de structures conditionnelles un affichage de la météo utilisant les capteurs de luminosité et le thermomètre	Comprendre et savoir utiliser les structures conditionnelles et les capteurs

TABLE 6.1 – Activités composant la séquence d'initiation à la programmation

6.2 Présentation des fiches

Chaque activité est décomposée en plusieurs fiches, réalisées en collaboration avec Anne Smal. Les fiches principales qui décrivent les activités et surnommées fiches "prof" et les fiches secondaires qui accompagnent les fiches principales, à savoir les fiches "Explicatives", les fiches "Élève" et les fiches "Aide".

6.2.1 Les fiches "prof"

Chaque fiche activité adopte la même structure décrivant point par point :

- « Compétences » décrit les compétences développées au cours de l'activité.
- « Objectifs » décrit ce que l'élève sera capable de faire en fin d'activité.
- « Description de l'activité » résume en quelques mots l'activité et ses objectifs.
- Liste des matériel et ressources nécessaires.
- Description de la préparation en vue de l'activité.
- « Déroulement », description étape par étape du déroulement de l'activité.
- « Solutions & Problèmes » montre une solution à l'activité ainsi qu'une explication pour aider les professeurs à comprendre si nécessaire.
- « Problèmes Techniques » informe le professeur d'éventuels problèmes techniques qu'il risque de rencontrer au cours de l'activité.

Tous ces points d'une fiche d'activité sont écrits et développés de façon à donner aux enseignants toutes les clés en main pour que l'activité se déroule correctement. Les fiches "prof" réalisées sont regroupées dans les annexes (A, B, C et D). La section "ressources nécessaires" liste notamment les fiches secondaires qui accompagnent cette activité.

6.2.2 Les fiches explicatives

La fiche explicative reprend les notions techniques liées à un concept, une fonctionnalité du micro :bit ou son interface. Elles sont à destination des enseignants qui n'auraient pas la formation nécessaire pour enseigner la matière présentée et leur servir d'aide afin d'apporter des réponses aux questions des élèves qui auraient des difficultés à réaliser une activité. Les fiches explicatives sont présentes en annexe (E, F, G et H).

6.2.3 Les fiches "élève"

La fiche "élève" est une fiche résumant les concepts vus à distribuer aux élèves en fin d'activité. Elle permet à l'élève d'avoir une trace de ce qu'ils ont appris et accompli au cours de l'activité et de résumer l'essentiel des concepts acquis en moins d'une page. En plus de ça, certaines de ces fiches proposent aux élèves d'aller plus loin avec des liens vers des vidéos de vulgarisation. Ces fiches sont regroupées dans les annexes (J, K, L et M).

6.2.4 Les fiches "aide"

La dernière ressource accompagnant certaines des fiches d'activités sont les fiches aides. À destination des élèves, elles leur servent d'aide en cas de difficulté au cours de l'activité et de pense-bête en leur rappelant les nouveaux blocs utiles à l'exercice et en présentant l'énoncé sous forme d'étapes. L'objectif de ces fiches est de ne pas perdre l'élève dans l'interface du micro:bit et de les recentrer

sur le plus important dans ses activités : l'acquisition et la compréhension des concepts. Ces fiches sont distribuées aux élèves en début d'activité et reprises à la fin. Elles ne sont présentes que pour les exercices aux consignes plus compliquées (activités 3 et 4). Elles sont présentes en annexe (O et P).

6.3 Présentation des activités

Les quatre activités de la séquence sont construites de la façon suivante :

6.3.1 Activité 1 : "Premiers pas avec le micro:bit – Découverte des entrées et sorties"

La première activité (annexe A) consiste, dans un premier temps, en une discussion entre professeur et élèves sur ce qu'est, pour eux, un ordinateur pour ensuite leur présenter le schéma fonctionnel de l'ordinateur comme établi précédemment (cfr Figure 4.1). L'objectif est de faire prendre conscience aux élèves que ce type de schéma est générique aux systèmes informatiques, y compris le micro:bit, qui n'est pas, par la majorité d'entre eux, considéré comme un ordinateur au premier regard. Cette partie se conclut par le brainstorming sur les différents périphériques (d'entrées et de sorties) d'un ordinateur. En seconde partie d'activité, par groupe de 3-4 élèves, il leur est proposé de découvrir, par la manipulation de micro:bit préprogrammés, les différents périphériques disponibles sur l'appareil ainsi que leur 'placement' dans le schéma fonctionnel ("entrée ou sortie?"). L'objectif dans cette première activité en débranché est de rendre les élèves actifs, de les impliquer directement dans la recherche et la compréhension de l'objet.

Chaque groupe manipule quatre micro:bit avec chacun un programme différent qui met en fonction des périphériques du micro:bit différents. Les micro:bit doivent être programmés au préalable par le professeur. Les programmes sont les suivants :

- microbit-bouton.hex : Ce code met en avant d'une part les différents boutons du micro:bit, à savoir A, B et A+B en affichant un symbole différent pour chacun d'eux, et d'autre part l'accéléromètre, en affichant un autre symbole lorsque l'on secoue le micro:bit. Ce programme a la particularité de mettre également en avant le buzzer comme périphérique de sortie en supplément de l'écran de LEDs.



FIGURE 6.1 – Programme microbit-bouton.hex

- microbit-boussole.hex : Ce programme met en avant la boussole en affichant une lettre en fonction de la direction vers laquelle pointe le micro:bit. Un 'N' est affiché s'il pointe vers le Nord, 'O' vers l'Ouest, etc.

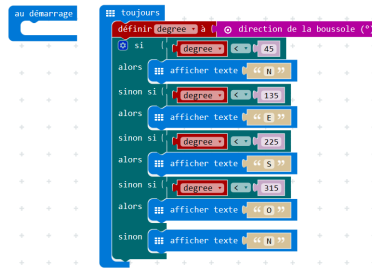


FIGURE 6.2 – Programme microbit-boussole.hex

- microbit-lumiere.hex : Ce programme exploite les capteurs de luminosité du micro:bit en affichant un chiffre différent en fonction du niveau de luminosité. En pressant le bouton A, le micro:bit va exploiter son thermomètre pour afficher la température ambiante.

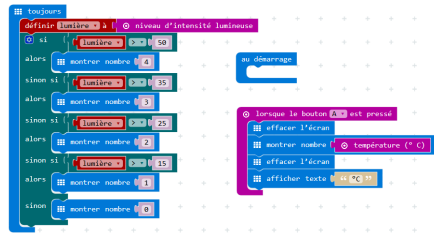


FIGURE 6.3 – Programme microbit-lumiere.hex

- microbit-accelerometre.hex : Ce dernier programme fait apparaître un point au centre de l'écran de LEDs du micro:bit que l'élève peut déplacer dans plusieurs directions à l'aide de l'accéléromètre.

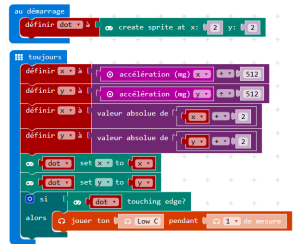


FIGURE 6.4 – Programme microbit-accelerometre.hex

La fiche explicative intitulée "Présentation du micro:bit" (annexe E) contient une description, à destination de l'enseignant, du micro:bit et de ses périphériques ainsi qu'une explication pour le branchement du buzzer. Une fiche "élève", "Découvertes des entrées et sorties" (annexe K), résume les concepts vus lors de l'activité (ordinateur, entrées, sorties, etc.) ainsi qu'une présentation du micro:bit. A la fin de cette fiche, est proposé un lien vers un épisode de l'émission éducative "C'est pas sorcier"

dédié à l'ordinateur : "Ordinateur, tout un programme"¹ afin d'inciter les élèves à se renseigner au-delà des activités proposées.

6.3.2 Activité 2 : "Premiers pas avec le micro:bit – Introduction à la programmation"

Après avoir manipulé une première fois le micro:bit, la seconde activité (annexe B) va consister en une initiation à la programmation de celui-ci via l'interface « officielle » de programmation en blocs associés. Sans que ne soient encore explicités les différents concepts de base, les élèves sont laissés en situation de création, se limitant à emboîter les blocs dits de base (principalement l'affichage). Ils doivent alors transférer leur premier programme sur le système tangible afin de le tester.

Une fiche explicative sur l'interface de l'application du micro:bit, "Introduction à l'application du micro:bit" (annexe F), accompagne l'activité afin de présenter en détail l'interface à l'enseignant. La Fiche "Elève", "Introduction à la programmation" (annexe L) résume le concept de programmation comme présenté dans le chapitre 5 ainsi que l'interface web du micro:bit. Afin de les inciter à découvrir la programmation au-delà de l'activité, un lien vers une vidéo didactique youtube intitulée "LET'S ALGO 1 – C'est quoi un algorithme?"², réalisée par la chaîne 'Flesh & Computers' est référencée en fin de fiche. Cette vidéo leur présentera plus en détails les concepts de programmation et d'algorithme.

6.3.3 Activité 3 : "Variable et accéléromètre"

Après avoir découvert le micro:bit et son interface web, la troisième activité (annexe C) propose aux élèves de réaliser un programme qui exploite conjointement un concept de programmation et un capteur du micro-ordinateur : les variables et l'accéléromètre.

Dans un premier temps, l'enseignant présente aux élèves le concept de variable de façon théorique puis de pratique en laissant les élèves manipuler un micro:bit avec un programme (microbit-UneVariable.hex) affichant un chiffre augmentant de 1 en appuyant sur le bouton B et diminuant de 1 en appuyant sur le bouton A. Cela permet de symboliser de façon ludique la manipulation d'une variable et la modification de son contenu.

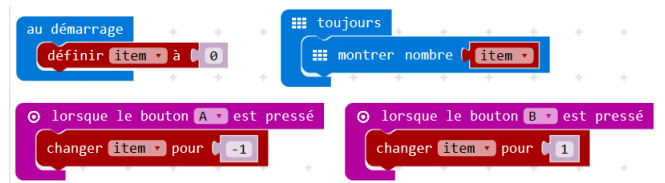


FIGURE 6.5 – Programme microbit-UneVariable.hex

Dans la seconde partie de l'activité, les élèves devront réaliser une calculatrice en créant deux variables initialisées à 0, dont la valeur augmentent de 1 lorsque l'on presse le bouton A ou B du micro:bit. L'addition des valeurs des deux variables s'effectue en penchant le micro:bit vers la gauche, leur soustraction

1. <https://www.youtube.com/watch?v=c96KP5jZVYkafin>, consultée en ligne le 20 août 2020.

2. <https://www.youtube.com/watch?v=I-5xj9jQQkk\&list=PLzRiCIzd4kGHJIaR7vZHY1lDBNrkJTgg-k>, consultée en ligne le 20 août 2020.

en penchant vers la droite, etc. Le résultat de l'opération s'affiche sur l'écran LED. Pour aller plus loin, les élèves peuvent ajouter des fonctionnalités à leur calculatrice comme, par exemple, la réinitialisation des variables à 0 en secouant l'appareil.

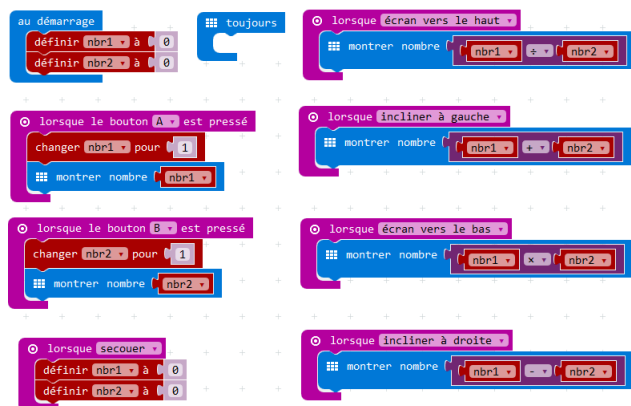


FIGURE 6.6 – Exemple de solution de l'activité "Variable et accéléromètre"

Afin de leur servir de guide durant l'exercice, une fiche "aide", "La calculatrice" (annexe P) permet de rappeler les consignes et les blocs utiles à la réalisation de l'objectif. Deux fiches explicatives accompagnent cette activité : "Les variables" (annexe G) et "L'accéléromètre du micro:bit" (annexe H) ainsi qu'une fiche "Elève" (annexe M) qui décrit le concept de variable et l'accéléromètre. Il propose aussi un lien vers la vidéo « LET'S ALGO 5 - Les affectations »³ de la chaîne Youtube 'Flech & Computers' qui explique de façon ludique le concept de variable.

6.3.4 Activité 4 : "Conditionnelle et boussole"

Lors de cette activité (annexe D), les conditionnelles seront présentées aux élèves qui devront les utiliser afin de réaliser un exercice qui exploite différents capteurs du micro:bit.

Après la présentation par l'enseignant des blocs correspondant aux conditionnelles et à la boussole, les élèves auront comme tâche de programmer un affichage météo qui affichera un soleil s'il fait beau (lumineux) et chaud, un nuage s'il fait chaud mais sombre et une lune dans les autres cas. Pour aller plus loin, ils pourront ajouter par la suite des fonctionnalités comme l'affichage d'un parapluie si il fait chaud et sombre ou l'affichage de la température en appuyant sur le bouton B.

3. <https://www.youtube.com/watch?v=71WinHtgc8A>, consultée en ligne le 20 août 2020.

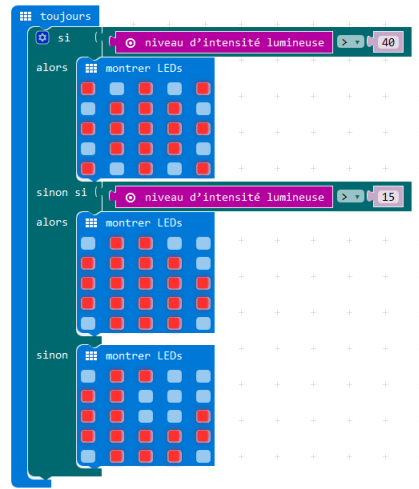


FIGURE 6.7 – Exemple de solution de l'activité "Conditionnelle et boussole"

Pour cet exercice, les élèves sont guidés par une fiche "Aide", "Affichage météo" (annexe P), qui rappelle, étape par étape, la réalisation du programme ainsi que les blocs nécessaires. Deux fiches explicatives, "Les conditionnelles" (annexe I) et "La boussole, le thermomètre et les capteurs de luminosité du micro:bit" (annexe J) sont disponible en support pour l'enseignant. Les élèves peuvent emporter les fiches "Elève" et "Les conditionnelles et les capteurs" afin de garder une trace de ce qu'ils ont appris.

Chapitre 7

Retours et résultats

Ce chapitre est dédié à la façon dont les activités ont été reçues. La première partie s'intéressera aux retours des élèves et la seconde, à l'analyse des résultats obtenus au moyen des pré- et post-tests décrits dans le chapitre 6. Le chapitre conclura par une discussion des résultats.

7.1 Retour des élèves

Les premiers contacts des élèves avec le micro:bit et de son interface ont été très positifs. Ils ont, de fait, montré beaucoup de curiosité et d'enthousiasme lors des périodes de manipulation des deux premières activités. Pendant la première activité, lors de la manipulation des micro:bit préprogrammés, ils ont particulièrement fait preuve de curiosité dans la découverte de ses différents périphériques. Durant la seconde activité, leur premiers pas sur l'interface web se sont faits avec beaucoup d'engouement même s'ils étaient moins à l'aise dans le transfert de leur programme vers le micro:bit qui nécessite de passer par les fichiers de l'ordinateur. Ce dernier souci peut cependant être évité en utilisant la version exécutable de l'interface qui transfère directement le programme vers le micro:bit.

Le fait que, dans le cadre de cette recherche, les activités aient été données par des informaticiens qu'ils ne connaissaient pas, a attisé la curiosité de certains et diminué l'attention pour les activités pour d'autres, ne considérant pas les informaticiens comme une figure d'autorité.

Pour les activités 3 et 4, la majorité des élèves sont arrivés à réaliser les programmes demandés. Il ressort une vraie volonté de la part des élèves de programmer, de tester sur le micro:bit et de chercher à corriger et améliorer leur programme. Mais si la manipulation des conditionnelles n'a pas posé de problème, il en a été différemment pour les variables. Si le concept semble avoir été bien compris, les élèves semblent avoir été moins à l'aise dans leur manipulation.

Au final, les élèves sortent ravis des activités, charmés par l'aspect singulier du micro:bit et l'usage intuitif de son interface.

7.2 Analyse des résultats

Afin de mesurer l'évolution de la vision de l'ordinateur et de l'informatique en général, les élèves ont répondu à un test contenant des questions ouvertes, au début et à la fin de la séquence d'activités. Une analyse de la fréquence des mots utilisés dans les réponses a été réalisée au moyen du logiciel

IraMuTeQ¹.

"Pour toi, l'informatique, c'est ..."

Leur vision générale de l'informatique semble inchangée après les activités. La présence du mot "ordinateur" dans les réponses à cette question est très légèrement en baisse dans le post-test (de 46% à 39%). On notera l'apparition des notions d'entrée et de sortie dans les réponses à la question dans les post-tests (7% pour les deux mots).

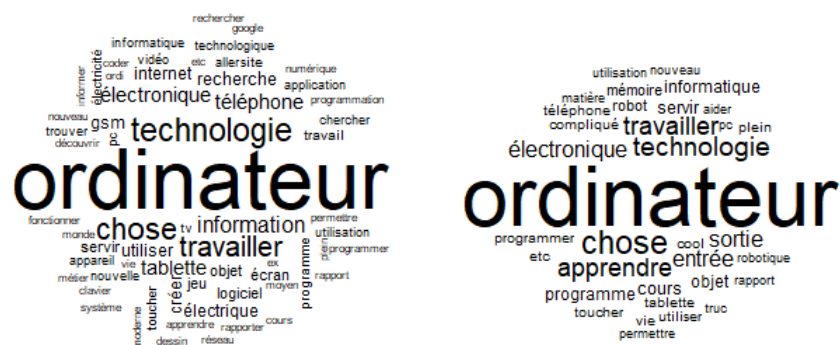


FIGURE 7.1 – Nuage de mot avant-après à la question "Pour toi, l'informatique, c'est ..."

"A quoi penses-tu quand tu entends 'métiers informatiques' ?"

Quand on analyse les réponses à la question sur leur vision des métiers informatiques, le mot ordinateur est ici aussi fortement présent aussi bien avant qu'après la séquence. Leur vision reste cependant centrée sur l'informaticien qui travaille sur son ordinateur.

1. <http://www.iramuteq.org/>, consultée en ligne le 20 août 2020.

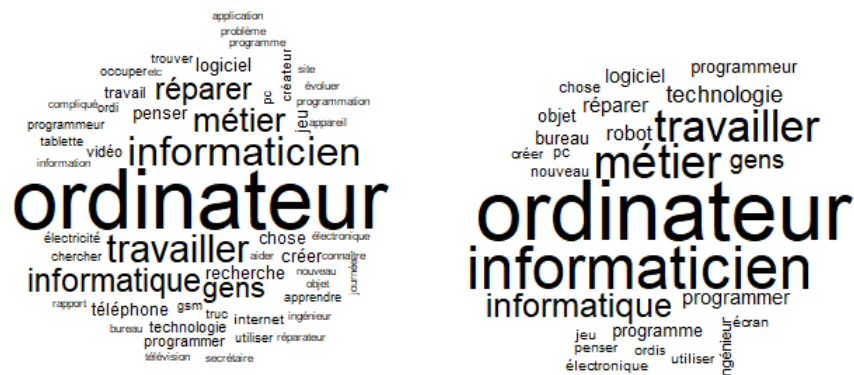


FIGURE 7.2 – Nuage de mot avant-après à la question
"À quoi penses-tu quand tu entends 'métiers informatiques'?"

"Selon toi, quels sont les points positifs/négatifs dans les 'métiers informatiques'?"

L'impact sur la vue du travail intensif sur écran reste, pour les élèves, un point négatif des métiers informatiques. Les résultats avant et après la séquence d'activité sont similaires. L'idée de découvrir et d'apprendre beaucoup de choses est, pour eux, le point le plus positif de ces métiers.

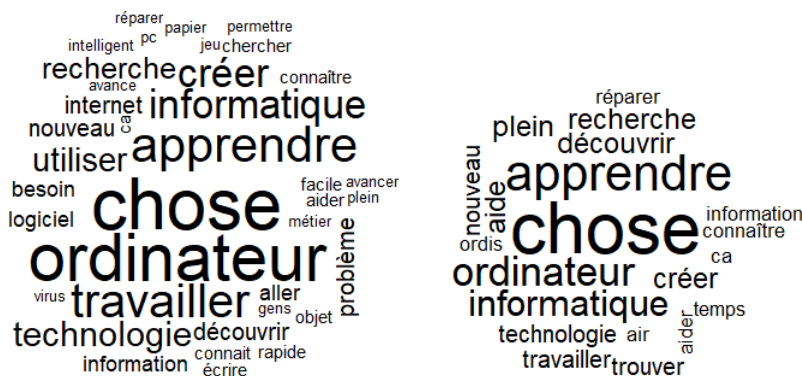


FIGURE 7.3 – Nuage de mot avant-après à la question
"Selon toi, quels sont les points positifs/négatifs dans les 'métiers informatiques'?"

"Selon toi, qu'est-ce qu'un ordinateur peut/ne peut pas faire ?

Si on constate un changement dans leur vision de l'ordinateur, celle sur les limites de l'ordinateur semble a priori inchangée. Au sujet de ce qu'un ordinateur peut faire, on retrouve avant comme après des réponses tournant autour de la recherche et de l'information. On notera cependant que le mot "internet" (de 16% d'occurrences dans les réponses des élèves à 6% en post-tests) est moins présent après la séquence [tandis que le mot "calculer" est légèrement plus présent] montrant qu'ils définissent plus un ordinateur comme une machine collectant et traitant de l'information qu'une machine connectée à internet.

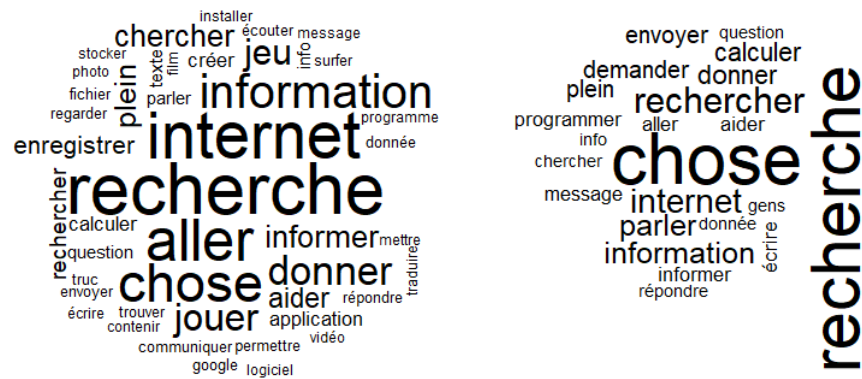


FIGURE 7.6 – Nuage de mot avant-après à la question
"Selon toi, qu'est-ce qu'un ordinateur peut faire ?"

Quant à ce qu'un ordinateur ne peut pas faire, les réponses sont assez similaires mais on constatera un renforcement des réponses "émotion" (8% de plus), "sentiment" (7% de plus) et "ressentir" (9% de plus).

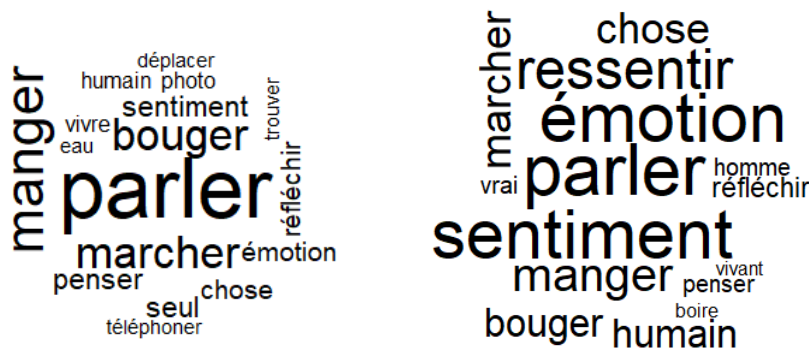


FIGURE 7.7 – Nuage de mot avant-après à la question
"Selon toi, qu'est-ce qu'un ordinateur ne peut pas faire ?"

Réponses au vrai ou faux

Pour le sondage vrai ou faux sur les préjugés, le choix a été fait de ne prendre en compte que les réponses des élèves ayant répondu aux pré- et post-tests. Ce sont donc les réponses de 133 élèves qui sont résumées dans le tableau ci-après :

Question	% de vrai avant	% de vrai après	Évolution
Travailler dans l'informatique, c'est toujours faire la même chose.	12%	14,3%	+ 2,3%
Travailler dans l'informatique, c'est travailler derrière un PC.	46,6%	40,6%	- 6%
Travailler dans l'informatique, c'est uniquement taper des lignes de code.	7,5%	13,5%	+ 6%
Les gens qui travaillent dans l'informatique n'ont pas de vie sociale.	6%	10,5%	+ 4,5%
Pour travailler dans l'informatique, il faut tout savoir sur les ordinateurs.	66,1%	55,6%	- 10,5%
Pour travailler dans l'informatique, il faut être surdoué(e) en maths.	19,5%	23%	+ 3,5%
L'informatique, c'est pour les hommes.	5%	9,7%	+ 4,7%

L'évolution la plus significative dans les résultats se situe dans la perception qu'ont les étudiants sur les connaissances techniques nécessaires pour la pratique de l'informatique. Chez plus de 10% des participants, l'idée que des connaissances approfondies soient nécessaires a changé. Ceci permet de penser que les outils utilisés pour l'expérience sont un facteur de simplification et de démystification des outils informatiques. Grâce à leur simplicité de conception et de programmation, l'expérience a prouvé aux étudiants qu'il était possible d'obtenir des résultats tangibles sans connaissances techniques poussées. À noter que l'expérience n'a pas eu d'incidence sur l'opinion relative aux connaissances en mathématique. Le cliché que l'informatique ne se pratique que derrière un PC a également légèrement évolué positivement à l'inverse de celui relatif à la programmation qui s'est renforcé dans une même proportion. Ces résultats peuvent paraître contradictoires mais ils peuvent s'expliquer par les outils utilisés lors de l'expérience. Si l'utilisation des cartes électroniques a bien démontré que l'informatique ne se limitait pas uniquement aux PC, leur programmation, parfois longue, a sans doute renforcé l'idée qu'elle était centrale dans l'informatique. Enfin, l'opinion que l'informatique est un univers répétitif, masculin et de geek, s'est très légèrement renforcée. Cette évolution limitée des résultats ne permet pas de dégager des tendances significatives.

Cinquième partie

Conclusion

Conclusion

Dans une société où le numérique prend une place de plus en plus importante dans notre quotidien, il devient nécessaire de former les plus jeunes générations au numérique afin qu'ils puissent appréhender ce monde de plus en plus connecté. L'une des étapes essentielles pour y parvenir est de les aider à démystifier l'ordinateur. C'est dans ce contexte que s'inscrit la séquence de quatre activités centrée autour de la découverte de l'ordinateur et de l'utilisation du micro:bit, ainsi que la volonté d'en montrer son efficacité auprès de jeunes élèves de 12-15 ans.

Au vu de ces premiers résultats, plusieurs objectifs peuvent être considérés comme atteints. En effet, après les activités, les élèves ont dorénavant une vision plus large de ce qu'est l'ordinateur ne se limitant plus à l'image limitée de l'ordinateur de bureau qu'ils avaient en début d'activité.

Le micro:bit semble avoir fait ses preuves. Sa particularité d'objet tangible lui permet de servir de support physique tant à la compréhension de concepts tels que le schéma fonctionnel d'un ordinateur et la programmation. Alliant les avantages des objets programmables tangibles et de la programmation par blocs, le micro-ordinateur s'est montré efficace en tant qu'aide à l'apprentissage du concept d'ordinateur, de son fonctionnement et de ses limites. À cela s'ajoute le fait que les élèves semblent prendre beaucoup de plaisir à le découvrir et le manipuler, que ce soit lors d'exercices débranchés ou de programmation.

Cependant, tous les résultats obtenus ne sont pas complètement positifs. Bien que l'on observe une légère évolution sur certaines questions, l'efficacité des activités sur la perception de l'informatique en général et des métiers informatique n'est pas complète puisque quelques préjugés persistent.

La recherche effectuée dans le cadre de ce mémoire a déjà mené à l'écriture d'un article (annexe Q) : "Comprendre l'ordinateur à travers un système informatique tangible, le micro:bit" (N.Théate, A.Smal, B.Fréney et J.Henry) ainsi qu'à un atelier (annexe R) "C'est quoi un ordinateur? Bien plus qu'un micro:bit ..." (N.Théate et J.Henry) dans le cadre du colloque scientifique Ludovia#CH 2018. Il s'agissait de la première édition de Ludovia Suisse et portait sur le thème "Émanciper l'école et la société avec le numérique?".

À l'avenir, la séquence d'activités ne demande qu'à s'améliorer et à s'adapter comme ça a été le cas avec l'ajout des fiches "aide". Plus les activités seront testées, plus elles évolueront de façon à être plus adaptées et seront donc plus efficaces. Pour les sujets où la différence des réponses est peu marquée, une séquence d'activités plus longue pourrait permettre de faire plus évoluer les opinions. Il serait intéressant de voir des enseignants utiliser les fiches ("Prof", "Explicative", "Elève" et "Aide") réalisées afin de mener à leur tour ces activités avec leurs élèves de manière autonome.

Bibliographie

- [1] Bruillard Eric Baron Georges-Louis. *Technologies de l'information et de la communication et « indigènes numériques » : quelle situation.*, volume 15. ÉPAL : Échanger Pour Apprendre en Ligne, 2008.
- [2] Stephanie Carretero, Riina Vuorikari, and Yves Punie. DigComp 2.1 : The Digital Competence Framework for Citizens with eight proficiency levels and examples of use. JRC Working Papers JRC106281, Joint Research Centre (Seville site), May 2017.
- [3] Seanpaul Gibson and Patrick Walton Bradley. A study of northern ireland key stage 2 pupils' perceptions of using the bbc micro :bit in stem education. 2017.
- [4] S. M. Glynn. *Making science concepts meaningful to students : Teaching with analogies. In S. Mikelskis-Seifert, U. Ringelband, & M. Brückmann (Eds.).* Münster : Waxmann, 2008.
- [5] F. Heintz, Linda Mannila, and Tommy Färnqvist. A review of models for introducing computational thinking, computer science and computing in k-12 education. *2016 IEEE Frontiers in Education Conference (FIE)*, pages 1–9, 2016.
- [6] Henry J. and Joris N. L'enseignement de l'informatique en belgique francophone : état des lieux. 2013.
- [7] Henry J. and Joris N. Informatics at secondary schools in the french-speaking region of belgium : myth or reality ? 2016.
- [8] N. Joris and J. Henry. L'enseignement de l'informatique en belgique francophone : état des lieux. In *Actes du colloque Didapro 5 - Dida&STIC (Sciences et technologies de l'information et de la communication en milieu éducatif)*, 2013. Didapro 5 - Dida&STIC ; Conference date : 28-10-2013 Through 30-10-2013.
- [9] Orni Meerbaum-Salant, Michal Armoni, and Mordechai Ben-Ari. Learning computer science concepts with scratch. *Computer Science Education*, 23 :239–264, 09 2013.
- [10] C O'Malley and D Stanton Fraser. Literature review in learning with tangible technologies. Workingpaper, FutureLab, 2004. Report 12 ID number : 0-9548594-2-1.
- [11] The Committee on European Computing Education (CECE). Informatics education in europe : Are we all in the same boat ? Technical report, New York, NY, USA, 2017.
- [12] David Weintrop and Uri Wilensky. Comparing block-based and text-based programming in high school computer science classrooms. *ACM Trans. Comput. Educ.*, 18(1), October 2017.

Annexe A

Fiche "prof" : Premiers pas avec le
micro:bit - Découverte des entrées et
sorties



Premiers pas avec le micro:bit - Découverte des Entrées et Sorties

Découverte - 1 période


Découverte - Entrée - Sortie

Compétences

Décrit les compétences développées (vis-à-vis du référentiel officiel).

Objectifs

L'élève sera capable de :

- Reconnaître les différents périphériques d'entrée et sortie d'un ordinateur 
- Manipuler le micro:bit




Description de l'activité

Cette activité est une introduction à la programmation et au micro:bit à des élèves n'ayant pas de connaissances en informatique.

Cela permettra aux élèves d'avoir une vision nouvelle de l'informatique et de ce que sont les périphériques d'entrées et de sorties d'un ordinateur.

Matériel et ressources

Matériel nécessaire :

- 1 micro:bit par élève 
- 1 batterie 
- 1 buzzer avec 2 pinces crocodiles 

Ressources nécessaires :

- Fiche explicative : "Présentation du micro:bit"
- Quatre codes fournis (microbit-bouton.hex, microbit-boussole.hex, microbit-lumiere.hex et microbit-accelerometre.hex)

Préparation

Pour l'activité, chacun des quatre codes doit être préinstallé sur cinq micro:bit (semblable si possible pour différencier les micro:bit ayant le même code)

Pour les micro:bit ayant le code "microbit-bouton.hex", il faut y brancher un buzzer. Pour cela, il faut relier, à l'aide des pinces croco, le fil noir à la broche 'GND' du micro:bit et le fil rouge à la broche portant le numéro 0.



Déroulement

Etape 1 : Introduction (15 min)

Qui	Fait quoi	Matériel
Le Professeur et les Elèves	Discutent de ce qu'est un ordinateur et brainstorment sur les différents périphériques de l'ordinateur	

Etape 2 : Préparation de l'activité (5 min)

Qui	Fait quoi	Matériel
Les Elèves	forment des groupes de quatre élèves	
Le Professeur	distribue quatre micro:bit par groupe, chaque micro:bit ayant un code différent	Les micro:bit avec les codes préinstallés, les batteries et les buzzers attachés

Etape 3 : Découverte du micro:bit (10 min)

Qui	Fait quoi	Matériel
L'Elève	Manipule les micro:bit pour trouver les différentes fonctionnalités des programmes et les périphériques d'entrée et sortie possibles du micro:bit	Les micro:bit, les batteries et les buzzers attachés

Etape 4 : Débriefing et Construction des Logigrammes (15 min)

Qui	Fait quoi	Matériel
Le Professeur et les Elèves	Mettent en commun leurs découvertes et construisent ensemble les logigrammes correspondants au programme	

Pour aller plus loin

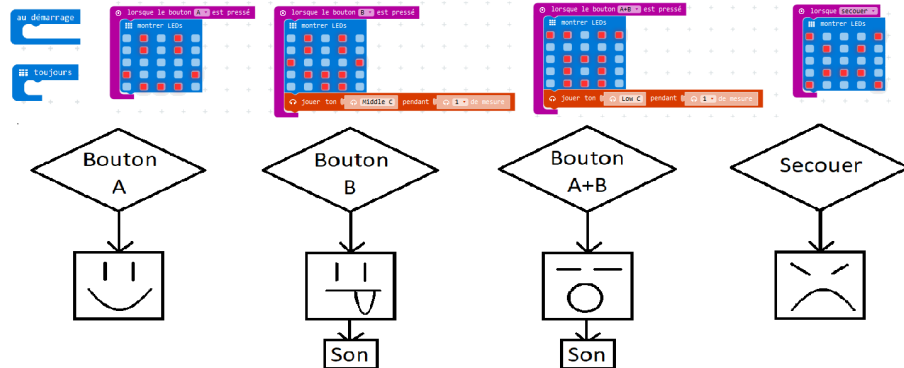
Les élèves peuvent, une fois avoir trouvé les différents périphériques d'entrée et sortie, réfléchir à différentes applications possibles utilisant les fonctionnalités offertes par ces périphériques.

Solutions & Problèmes

Pour allumer un micro:bit, il suffit d'y brancher la batterie . Cela lancera automatiquement le programme se trouvant dessus. Les quatres codes se présentent ainsi :

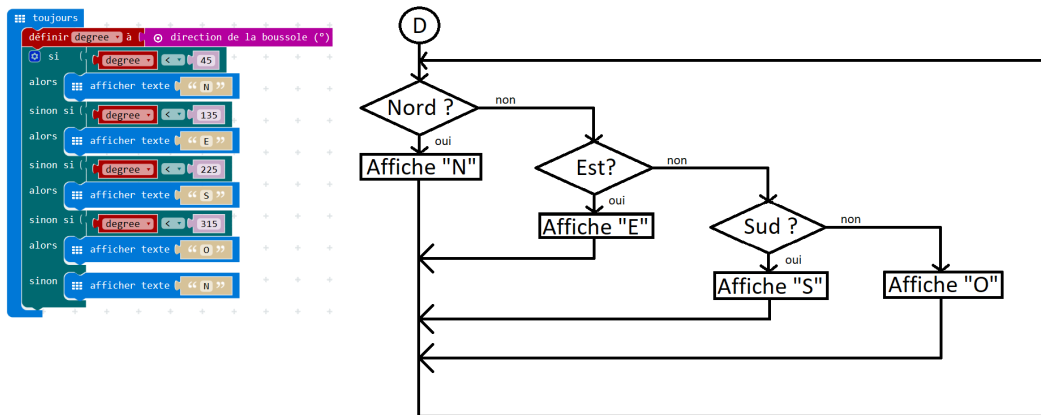
- microbit-boutton.hex

Ce code va faire apparaître une image ou faire un son différent en fonction du bouton pressé ou si l'utilisateur secoue le micro:bit



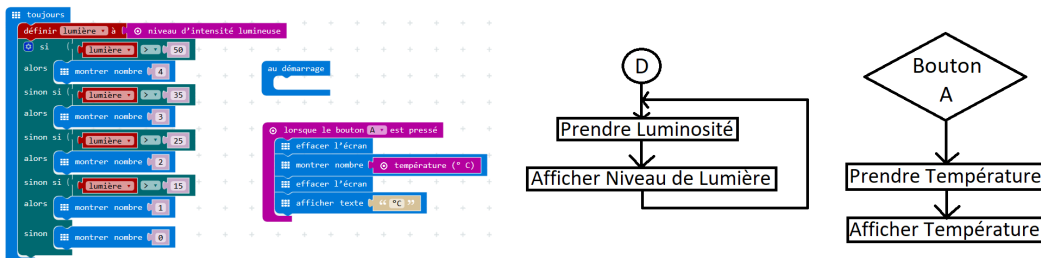
- microbit-boussole.hex

Ce code transforme le micro:bit en boussole et va faire apparaître un lettre en fonction de la direction vers laquelle point le micro:bit. Un 'N' si il pointe vers le Nord, 'E' vers l'Est, et ect.



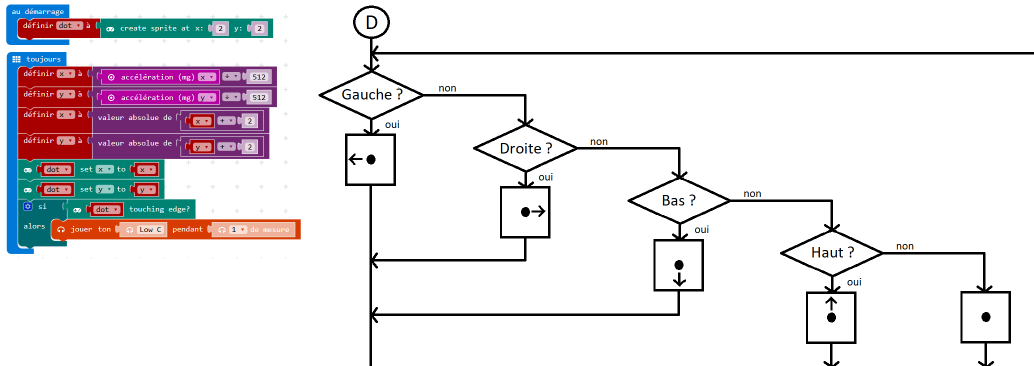
- microbit-lumière.hex

Ce code affiche un niveau de lumière en fonction de la luminosité (un 4 correspond à un environnement lumineux et un 0 à un environnement très sombre) mesurée à l'aide des capteurs de luminosité. En appuyant sur A, le programme affiche la température récupérée en utilisant le thermomètre.



- microbit-accelerometre.hex

Ce code fera apparaître un point au centre de l'écran. L'utilisateur pourra utiliser l'accéléromètre pour déplacer le point dans plusieurs directions.

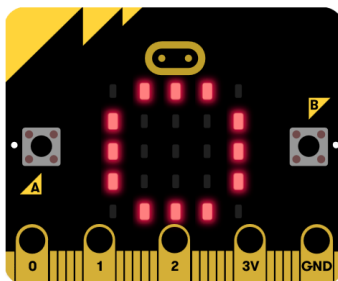


Problème technique

Si un programme ne se lance pas correctement,
Appuyer sur le bouton 'Reset' au dos de l'appareil pour relancer le programme

Si un programme ne s'exécute pas correctement (affichage étrange, son particulier, ...)
Vérifier bien que les pinces crocodile servant à brancher le Buzzer sont bien brancher sur les broches portant le numéro '0' et le signe 'GND' sans déborder sur les broches adjacentes. (cfr. la fiche explicative "Présentation du micro:bit" pour plus d'informations sur le branchement du Buzzer)

Dans le cas où le micro:bit affiche le message 'DRAW A CIRCLE' sur l'écran,
Cela signifie que le micro:bit doit calibrer la boussole. Il faut alors déplacer le point qui se trouve au centre de l'écran en bougeant le micro:bit vers la gauche, la droite, le haut et le bas afin de dessiner un cercle. Une fois cela fait, le programme se lancera.



Annexe B

Fiche "prof" : Premiers pas avec le micro:bit - Introduction à la programmation

Premiers pas avec le micro:bit - Introduction à la programmation



Programmation par bloc - 1 période
Découverte - Programmation - Entrées - Sorties

Compétences

Décrit les compétences développées (vis-à-vis du référentiel officiel).

Objectifs

L'élève sera capable de :




- Manipuler le micro:bit
- Transférer un programme vers le micro:bit
- Utiliser l'application web du micro:bit 
- Réaliser un programme utilisant les boutons et l'écran du micro:bit 

Description de l'activité

Cette activité est une introduction à la programmation pour des élèves n'ayant pas de connaissance en informatique. Cette introduction se fera grâce à l'application web de programmation par bloc du micro:bit.

Cela permettra aux élèves d'avoir une vision nouvelle de l'informatique.

Matériel et ressources

Matériel nécessaire :	Ressources nécessaires :
<ul style="list-style-type: none"> • 1 micro:bit par élève  • 1 câble microUSB par élève  • 1 ordinateur par élève • 1 buzzer avec 2 pinces crocodiles par élève  	<ul style="list-style-type: none"> • Fiche explicative - Présentation du micro:bit • Fiche explicative - Introduction à l'interface du micro:bit

Préparation

Allumer à l'avance les ordinateurs sur l'application web de micro:bit (<https://makecode.microbit.org/>) et brancher chaque buzzer à un micro:bit.



Déroulement

Etape 1 : Présentation de l'application web du micro:bit (15 min)

Qui	Fait quoi	Matériel
Le professeur	Présente l'interface du site du micro:bit, ses différentes sections et différentes catégories de blocs d'instruction	Un ordinateur et la fiche "Introduction à l'application web"
L'élève	Parcourt les différentes sections de l'interface pour comprendre leurs fonctions	Un ordinateur

Etape 2 : Préparation de la première activité (5 min)

Qui	Fait quoi	Matériel
Le professeur	Distribue un micro:bit, un câble micro USB, un buzzer et deux pinces crocodile par élève et présente l'activité à effectuer	Les micro:bit avec buzzer et les câbles micro USB
L'élève	Branche son micro:bit à son ordinateur à l'aide du câble micro USB	Un micro:bit avec buzzer, un câble micro USB et un ordinateur

Etape 3 : Première activité (20 min)

Qui	Fait quoi	Matériel
L'élève	Utilise l'application pour pouvoir faire apparaître certains symboles ou faire sonner certains sons avec le buzzer en fonction d'une action (comme appuyer sur un bouton, secouer le micro:bit, le faire pencher sur la gauche, ect.)	Un micro:bit avec buzzer, un câble micro USB et un ordinateur

Etape 4 : Conclusion (5 min)

Qui	Fait quoi	Matériel
Le professeur et les élèves	mettent en commun leurs découvertes et débriefent sur ce qu'ils ont vu au cours de l'activité	

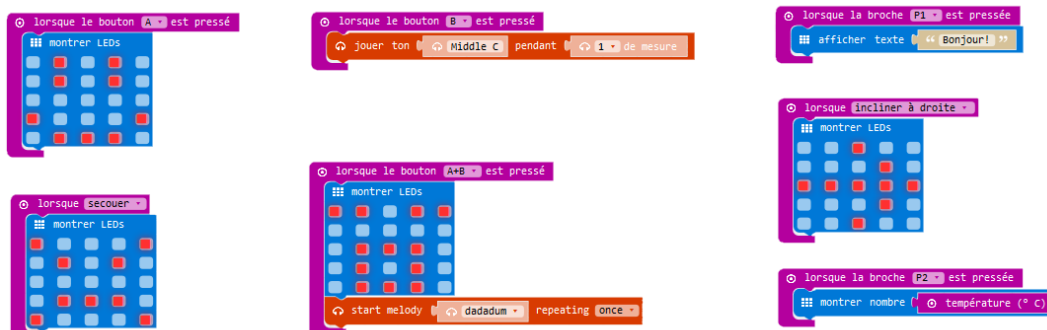
Pour aller plus loin

Si les élèves ont bien compris comment faire un programme qui utilise les boutons, le buzzer et l'écran, ils peuvent commencer à exploiter d'autres périphériques d'entrée comme les capteurs de luminosité, l'accéléromètre ou le thermomètre.



Solutions & Problèmes

L'élève est libre dans la résolution de l'activité mais voici quelques exemples de solution :



Les blocs bleus correspondent à des affichages simples (comme l'affichage de nombre, message ou symboles). Les blocs oranges sont pour les sons sortant du buzzer (sons simples ou musiques). Les blocs mauves sont pour les entrées qui déclencheront une action.

Problème technique

Si un programme ne se lance pas correctement,
Appuyer sur le bouton 'Reset' au dos de l'appareil pour relancer le programme

Si un programme ne s'exécute pas correctement (affichage étrange, son particulier, ...)
Vérifier bien que les pinces crocodile servant à brancher le Buzzer sont bien brancher sur les broches portant le numéro '0' et le signe 'GND' sans déborder sur les broches adjacentes. (cfr. la fiche explicative "Présentation du micro:bit" pour plus d'informations sur le branchement du Buzzer)

Annexe C

Fiche "prof" : Variable et accéléromètre



Variable et Accéléromètre



Programmation par bloc - 1 période
Programmation - Variable - Capteurs - Découverte

Compétences

Décrit les compétences développées (vis-à-vis du référentiel officiel).

Objectifs

L'élève sera capable de :

- Manipuler le micro:bit
- Comprendre le fonctionnement des variables 
- Réaliser un programme utilisant l'accéléromètre du micro:bit 



Description de l'activité

Cette activité est une introduction à la programmation pour des élèves s'étant déjà familiarisé avec l'application web du micro:bit.

Cela permettra aux élèves d'avoir une vision nouvelle de l'informatique et d'exploiter des possibilités offertes par différents capteurs comme l'accéléromètre.

Matériel et ressources

Matériel nécessaire :

- 1 micro:bit par élève 
- 1 câble microUSB par élève 
- 1 ordinateur par élève
- 1 projecteur

Ressources nécessaires :

- Fiche explicative - Introduction à l'interface du micro:bit
- Fiche explicative - L'accéléromètre du micro:bit
- Fiche explicative - Les variables
- Le code microbit-UneVariable.hex

Préparation

Allumer à l'avance les ordinateurs sur l'application web de micro:bit (<https://makecode.microbit.org/>).
Le programme, microbit-UneVariable.hex, doit être préinstallé sur les micro:bit.

Déroulement

Etape 1 : Présentation de l'application web (10 min)

Qui	Fait quoi	Matériel
Le professeur	Présente les différents blocs (/instructions) correspondant aux variables et à l'accéléromètre	Un ordinateur, un projecteur et les fiches "L'accéléromètre du micro:bit" et "Les variables"

Etape 2 : Préparation de l'activité (5 min)

Qui	Fait quoi	Matériel
Le professeur	Distribue un micro:bit, un câble micro USB par élève et présente l'activité à effectuer	Les micro:bit et les câbles micro USB
L'élève	Branche son micro:bit à son ordinateur à l'aide du câble micro USB	Un micro:bit, un câble micro USB et un ordinateur

Etape 3 : Manipulation d'une variable (5 min)

Qui	Fait quoi	Matériel
L'élève	Découvre la manipulation de la valeur d'une variable en augmentant ou diminuant la valeur affichée par le micro:bit en pressant les boutons A et B	Un micro:bit, un câble micro USB et un ordinateur

Etape 4 : Programmation de la calculatrice (20 min)

Qui	Fait quoi	Matériel
L'élève	Crée un programme avec : - Deux variables, initialisées à 0. - La valeur de la première variable augmente de 1 en appuyant sur le bouton A - La valeur de la seconde variable augmente de 1 en appuyant sur le bouton B - Le micro:bit affiche l'addition des deux valeurs en penchant l'appareil à gauche et la soustraction en penchant à droite	Un micro:bit, un câble micro USB et un ordinateur

Etape 5 : Conclusion (5 min)

Qui	Fait quoi	Matériel
Le professeur et les élèves	mettent en commun leurs découvertes et débriefent sur ce qu'ils ont vu au cours de l'activité	

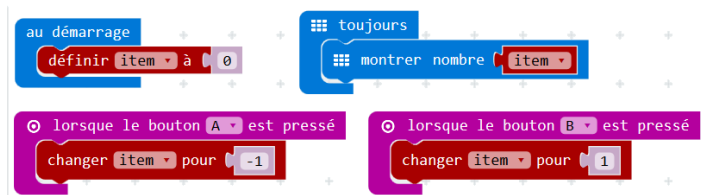
Pour aller plus loin

Si les élèves ont bien compris comment exploiter les possibilités offertes par les variables et l'accéléromètre, ils peuvent rajouter de nouvelles fonctionnalités comme : - Remise à 0 des deux variables en secouant le micro:bit. - Ajouter la multiplication et la division en penchant vers d'autres directions. Après cela, ils peuvent réfléchir à différentes façons d'exploiter l'accéléromètre d'autres manières.

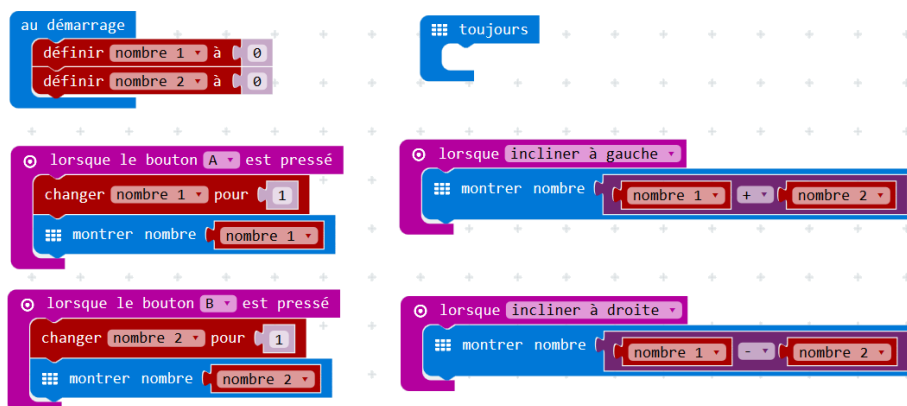


Solutions & Problèmes

Le code fourni, microbit-UneVariable.hex, se présente ainsi :

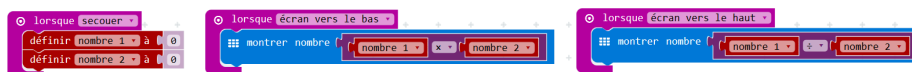


L'élève est libre dans la résolution de l'activité mais voici un exemple de solution :



On initialise deux variables (les noms des variables importent peu mais l'idéal est qu'ils soient significatif) au lancement du programme à 0. On augmente la valeur de la première variable de 1 en appuyant sur le bouton A et de la seconde en appuyant sur B. On va afficher le résultat de l'addition en penchant à gauche le micro:bit, la soustraction en penchant à droite.

On peut rajouter des fonctionnalités comme la remise à 0 des valeurs ou l'ajout de la multiplication et de la division comme pour le code suivant :



Annexe D

Fiche "prof" : Conditionnelle et boussole



Conditionnelle et Boussole



Programmation par bloc - 1 période
Programmation - Conditionnelle - Capteurs - Découverte

Compétences

Décrit les compétences développées (vis-à-vis du référentiel officiel).

Objectifs

L'élève sera capable de :

- Manipuler le micro:bit
- Comprendre le fonctionnement des conditionnelles 
- Réaliser un programme utilisant la boussole du micro:bit 



Description de l'activité

Cette activité est une introduction à la programmation pour des élèves s'étant déjà familiarisé avec l'application web du micro:bit.

Cela permettra aux élèves d'avoir une vision nouvelle de l'informatique et d'utiliser d'exploiter des possibilités offertes par différents capteurs.

Matériel et ressources

Matériel nécessaire :

- 1 micro:bit par élève 
- 1 câble microUSB par élève 
- 1 ordinateur par élève
- 1 projecteur

Ressources nécessaires :

- Fiche explicative - Introduction à l'interface du micro:bit
- Fiche explicative - La boussole, le thermomètre et les capteurs de luminosité du micro:bit
- Fiche explicative - Les conditionnelles

Préparation

Allumer à l'avance les ordinateurs sur l'application web de micro:bit (<https://makecode.microbit.org/>).
Si il le faut, il faut recalibrer la boussole du micro:bit (voir page 3, à "Problème technique").

Déroulement

Etape 1 : Présentation de l'application web (15 min)

Qui	Fait quoi	Matériel
Le professeur	Présente les différents blocs (/instructions) correspondant au conditionnelle et à la boussole	Un ordinateur, un projecteur et les fiches "La boussole, le thermomètre et les capteurs de luminosité du micro:bit" et "Les conditionnelles"

Etape 2 : Préparation de l'activité (5 min)

Qui	Fait quoi	Matériel
Le professeur	Distribue un micro:bit et un câble micro USB par élève et présente l'activité à effectuer	Les micro:bit et les câbles micro USB
L'élève	Branche son micro:bit à son ordinateur à l'aide du câble micro USB	Un micro:bit, un câble micro USB et un ordinateur

Etape 3 : Programmation de la boussole (20 min)

Qui	Fait quoi	Matériel
L'élève	Crée un programme qui affiche les quatre points cardinaux en fonction de la direction vers laquelle pointe le micro:bit	Un micro:bit, un câble micro USB et un ordinateur

Etape 4 : Conclusion (5 min)

Qui	Fait quoi	Matériel
Le professeur et les élèves	mettent en commun leurs découvertes et débrièfent sur ce qu'ils ont vu au cours de l'activité	

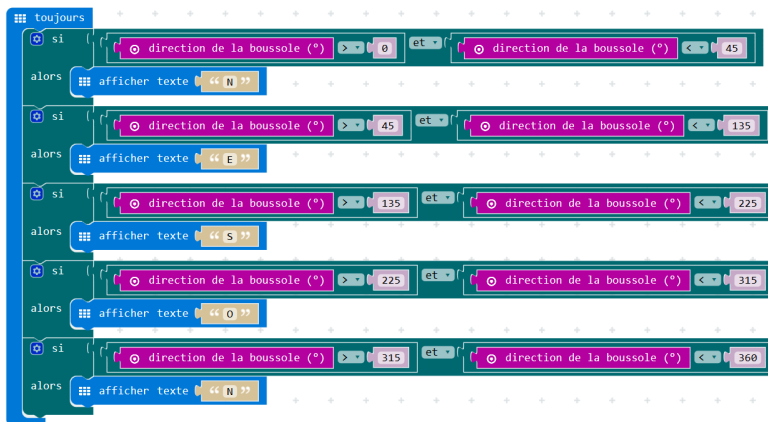
Pour aller plus loin

Si les élèves ont bien compris comment exploiter les possibilités offertes par les conditionnelles, ils peuvent essayer de faire une boussole plus précise qui afficherait les points cardinaux "Nord-Est", "Sud-Est", "Nord-Ouest" et "Sud-Ouest".

Ils peuvent aussi essayer de faire un affichage météo en utilisant les capteurs de lumière et le thermomètre. Ce dernier affichera un soleil s'il fait beau et chaud, un nuage s'il fait bon mais sombre ou une lune sinon. On définira qu'il fait beau si la luminosité est au dessus de 40, clair si elle est au dessus de 15 et qu'il fait chaud si la température est au dessus de 15 (on peut aussi demander d'afficher la température en appuyant sur un bouton).

Solutions & Problèmes

L'élève est libre dans la résolution de l'activité mais voici un exemple de solution :



Si la valeur de la boussole est entre 0° et 45°
ALORS on se dirige vers le Nord

Si la valeur de la boussole est entre 45° et 135°
ALORS on se dirige vers le Est

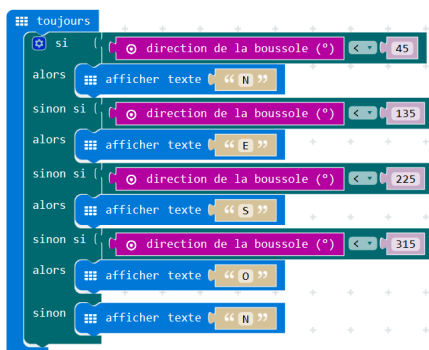
Si la valeur de la boussole est entre 135° et 225°
ALORS on se dirige vers le Sud

Si la valeur de la boussole est entre 225° et 315°
ALORS on se dirige vers le Ouest

Si la valeur de la boussole est entre 315° et 360°
ALORS on se dirige vers le Nord

La boussole renvoyant une valeur entre 0 et 360. On va définir quatres écarts pour désigner les quatres points cardinaux. Entre 45 et 135, on pointe vers l'Est. Entre 135 et 225, on pointe vers le Sud. Etc.

En cliquant sur l'engrenage du bloc 'IF', il est possible de créer des blocs 'IF...THEN ... ELSE IF...' permettant de créer un code plus clair et plus simple comme ceci :



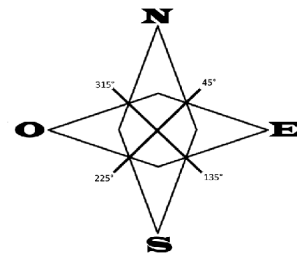
Si la valeur de la boussole est en dessous de 45°,
ALORS on se dirige vers le Nord.

Si la valeur de la boussole est en dessous de 135°,
ALORS on se dirige vers l'Est.

Si la valeur de la boussole est en dessous de 225°,
ALORS on se dirige vers le Sud.

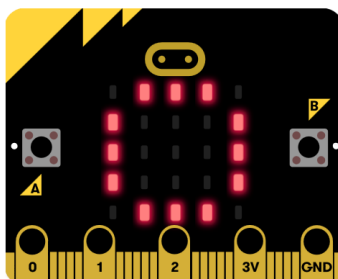
Si la valeur de la boussole est en dessous de 315°,
ALORS on se dirige vers l'Ouest

SINON, on se dirige vers le Nord



Problème technique

Dans le cas où le micro:bit affiche le message 'DRAW A CIRCLE' sur l'écran, Cela signifie que le micro:bit doit calibrer la boussole. Il faut alors déplacer le point qui se trouve au centre de l'écran en bougeant le micro:bit vers la gauche, la droite, le haut et le bas afin de dessiner un cercle. Une fois cela fait, le programme se lancera.



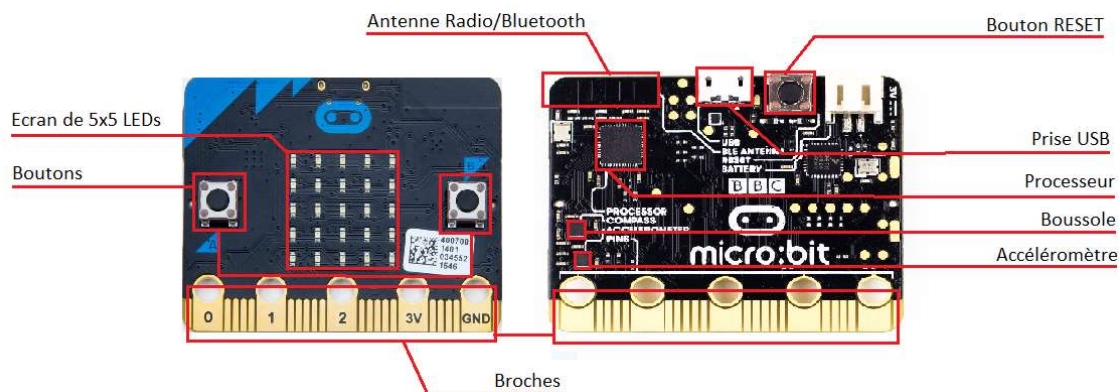
Annexe E

Fiche explicative : Présentation du micro:bit

Présentation du micro:bit

Le micro:bit est micro-ordinateur fort bien équipé permettant de nombreuses possibilités. Ce dernier n'a pas de batterie intégrée, il faut donc soit le brancher à une batterie, soit le brancher à une prise USB (comme sur un ordinateur) pour allumer un micro:bit. Une fois branché, le micro:bit exécutera le programme qu'il contient.

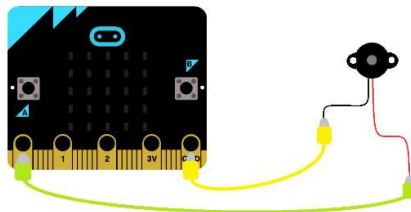
Il se présente ainsi :



En plus de cela, on retrouve un troisième bouton sous la forme du bouton 'A + B', des capteurs de luminosité sous l'écran de LEDs et une fonctionnalité permettant aux micro:bits de communiquer entre eux.

On retrouve aussi des broches sur le bas de l'appareil. Elles fonctionnent comme un circuit électrique, si l'utilisateur pince la broche marquée du numéro '0' et la broche correspondant au 'Ground' (soit la broche 'GND'), alors le micro:bit reconnaîtra la présence d'un signal électrique dans la broche '0'.

Il est possible de brancher d'autres périphériques au micro:bit grâce aux broches comme par exemple, un buzzer. Il se branche ainsi : à l'aide de deux pinces crocodiles, on branche le câble négatif (noir) sur la broche 'GND' et le câble positif (rouge) sur la broche portant le numéro '0'.



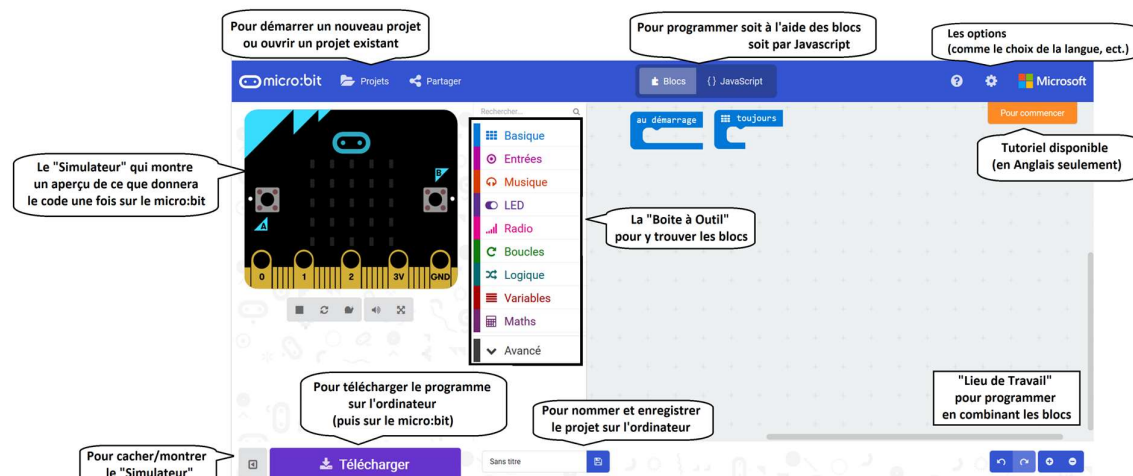
Pour modifier le code contenu dans l'appareil, il faut le brancher à un ordinateur par USB et mettre un nouveau programme comme pour glisser un fichier dans une clé USB.

Annexe F

Fiche explicative : Introduction à l'application du micro:bit

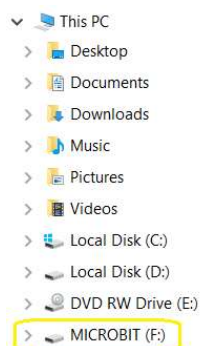
Introduction à l'application du micro:bit

Le micro:bit est un micro-ordinateur qui va exécuter le programme qu'il contient. Il est possible de faire son propre programme à l'aide du site officiel de micro:bit (<https://makecode.microbit.org/>) qui se présente ainsi :



Pour coder, il suffit de prendre des blocs (correspondant à une instruction/une action) dans la « Boîte à Outil » et de les combiner entre eux dans le « Lieu de Travail » comme pour un puzzle, on peut ensuite observer le résultat sur le « Simulateur ». Les différents blocs/instructions se décomposent en plusieurs catégories caractérisées par un nom et une couleur.

Des détails sur les instructions sont fournis sous la forme d'une description de l'instruction qui apparaît en laissant la souris sur un bloc. A noter aussi qu'un bloc de couleur grisâtre sur le « Lieu de Travail » est un bloc mal placé. Une fois satisfait du code, on peut le renommer et télécharger le fichier.



Une fois téléchargé, le fichier se trouvera dans le dossier « Téléchargement » (ou « Download ») et aura comme nom « microbit-<nom du programme>.hex ». Il suffit ensuite de le copier sur le micro:bit, au bon endroit comme pour une clé USB (voir ci-contre).

Le programme se chargera et se lancera alors automatiquement sur le micro:bit. On peut relancer le programme depuis le début en appuyant sur la touche RESET se trouvant au dos du micro:bit.

Cependant, le micro:bit ne peut contenir qu'un seul code à la fois. Dès que vous le branchez (soit par USB à un PC, soit en y branchant une batterie/des piles), il lance automatiquement le programme qu'il contient. Lorsque l'on copie un programme sur le micro:bit, le programme précédent est automatiquement supprimé, veillez à bien conserver vos programmes dans un fichier à part si vous souhaitez les conserver.

Annexe G

Fiche explicative : Les variables

Les variables

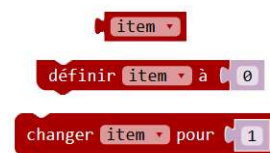
Etant donné qu'un programme/ordinateur traite beaucoup d'informations, il est parfois nécessaire de stocker certaines données quelque part. Pour stocker des données (par ex, le score dans un jeu vidéo qui évolue ou retenir le nom d'un utilisateur) que l'on va manipuler et traiter, le programmeur utilise des variables. Et certaines valeurs doivent être stockées sans pour autant être changer, ce sont les constantes (par ex, la valeur 'pi').

Une variable, c'est donc comme une petite boîte dans la mémoire, portant un nom, et dans laquelle on va y mettre des informations que l'on va réutiliser quand on aura besoin de son contenu. On peut y stocker des nombres mais aussi des suites de lettre.

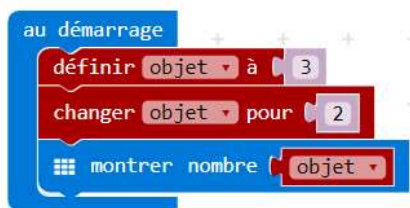
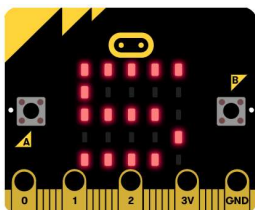
Ces valeurs seront stockées dans la mémoire de l'ordinateur. Quand un programme a besoin d'une valeur, il va aller chercher la variable correspondante dans la mémoire.

Étape	Instruction	Mémoire
1	a = 5;	a: 5
2	b = 2;	a: 5 b: 2
3	a = a + 3;	a: 8 b: 2
4	c = a * b;	a: 8 b: 2 c: 16
5	a = c;	a: 16 b: 2 c: 16
6	d = "Val";	a: 16 b: 2 c: 16 d: "Val"

En ce qui concerne le micro:bit, tout ce qui est relatif au variable se trouve dans la catégorie « variable ». On y trouve les blocs correspondant à nos variables que l'on peut attacher à des instructions pour afficher leurs contenus par exemple. On y trouve une instruction pour définir son contenu ainsi qu'une instruction qui va modifier son contenu en ajoutant ou en retirant un certain nombre (il faut que le contenu de la variable soit un nombre).



Par exemple, le programme suivant initialise une variable, portant le nom 'objet', à 15. Augmente la valeur de cette variable de 5 et affiche enfin le contenu de cette variable. (Il affichera 20 sur le microbit à la fin de ce programme) :



Annexe H

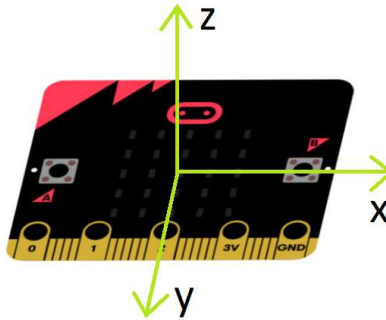
Fiche explicative : L'accéléromètre du micro:bit

L'accéléromètre du micro:bit

Un accéléromètre est un outil permet de mesurer les mouvements d'un objet. Traditionnellement, un accéléromètre fonctionne de manière mécanique à l'aide d'un gyroscope qui renvoie un signal électrique en fonction des mouvements effectués. Maintenant, on retrouve les accéléromètres sous forme de puce électroniques qui effectue le même travail.

On retrouve cette puce dans les smartphones par exemple pour pouvoir orienter correctement le sens de l'écran en fonction de l'orientation de l'appareil ou pour servir de podomètre, en comptant les pas.

C'est cette puce qui reconnaitra les moments où l'on secoue le micro:bit. On peut donc utiliser cet outil pour faire une alarme qui alertera l'utilisateur si quelqu'un bouge un objet sans accord par exemple.



Elle fonctionne de la façon suivante : l'accéléromètre mesure trois données selon les axes x, y et z. Au repos, donc posé à plat sur une table, l'accéléromètre renvoie comme donnée les valeurs : $x = 0$, $y = 0$ et $z = -1023$. On peut récupérer ces valeurs individuellement avec une instruction proposée par micro:bit :

accélération (mg) x

Et, que pour les boutons ou lorsque l'on secoue le micro:bit, on peut faire un code qui s'exécutera uniquement lorsque l'on bouge le micro:bit en avant, en arrière, à gauche et à droite :

lorsque écran vers le haut
lorsque écran vers le bas
lorsque incliner à gauche
lorsque incliner à droite

Annexe I

Fiche explicative : Les conditionnelles

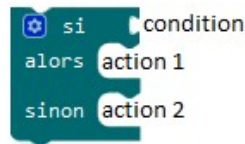
Les Conditionnelles

Afin de traiter au mieux certaines données, on va utiliser les instructions conditionnelles car une importante partie de la programmation consiste à définir 'quand' un programme doit effectuer une certaine tâche.

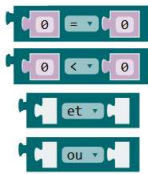
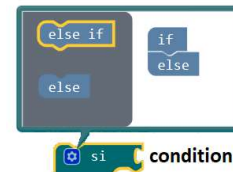
Dans une instruction conditionnelle, il faut définir une condition qui a pour valeur 'vrai' ou 'faux'. Si cette condition est 'vraie', alors l'instruction va exécuter l'« action 1 » mais si la condition est 'fausse', alors l'instruction va exécuter l'« action 2 ». On n'exécutera soit que l'« action 1 », soit l'« action 2 » en fonction de la condition.

Par exemple, c'est pareil à « SI il pleut, ALORS prends ton parapluie SINON tu peux aller jouer au Tennis » ou en mathématique : « SI $x < 5$ ALORS x est plus petit que 5 SINON x est plus grand que 5 ».

Pour ce qui est de la programmation par bloc du micro:bit, une instruction conditionnelle, ou instruction 'IF' en anglais, se présente avec ce bloc-ci, se trouvant dans la catégorie « logique » de la « boîte à outil » :



A noter que en cliquant sur l'engrenage se trouvant sur le bloc, on a accès à un sous-menu qui permet de modifier la forme de l'instruction en rajoutant de nouvelles conditions ou en supprimant le 'sinon', etc.... Cela peut-être utile si l'on souhaite être plus précis, par exemple : « SI $x \neq 5$ ALORS, SI $x < 5$ ALORS x est plus petit que 5 SINON x est plus grand que 5. »



Pour exploiter au mieux les possibilités des instructions conditionnelles, on trouve d'autres blocs dans la catégorie « Logique » comme les symboles 'égal', 'différent', 'plus grand que', etc... On y trouve aussi les blocs 'et' et 'ou' qui permette de prendre en compte plusieurs conditions (par exemple : «SI il pleut ET qu'on est Vendredi, ALORS prend ton parapluie »).

Annexe J

Fiche explicative : La boussole, le thermomètre et les capteurs de luminosité du micro:bit

La boussole, le thermomètre et les capteurs de luminosité du micro:bit

La boussole, le thermomètre et les capteurs de luminosité sont des capteurs présents sur le micro:bit. On peut récupérer leur valeur à l'aide des blocs suivants :

- La boussole

La boussole est un outil qui détecte les champs magnétiques comme le champ magnétique de la Terre et permet ainsi de repérer le Pôle Nord. Elle fonctionne donc moins bien entourer d'objets métalliques. La boussole renvoie une valeur comprise entre 0° et 360°. 0° correspond au Nord, 90° à l'Est, 180° au Sud et 270° à l'Ouest.

direction de la boussole (°)

- Le thermomètre

La valeur de la température est récupérée en Celsius. Le capteur de température est très sensible. Il arrive que la valeur qu'il reçoit soit supérieure à la réalité dû au fait que le processeur du micro:bit chauffe pour les utilisations à longue durée.

température (° C)

- Les capteurs de luminosité

La valeur du niveau d'intensité lumineuse, quant à elle, varie entre 0 et 255 mais en journée ensoleillée, la valeur tourne plus autour des 40-50. A savoir que les capteurs de luminosité se situent sur le micro:bit en dessous des LEDs.

niveau d'intensité lumineuse

On peut attacher ces blocs à d'autres instructions pour pouvoir afficher leurs valeurs sur l'écran par exemple, comme ceci :

montrer nombre niveau d'intensité lumineuse

On peut aussi stocker la valeur de ces capteurs dans une variable pour, par exemple, effectuer une action en fonction de la valeur d'un capteur comme afficher un soleil si il fait plus de 20° :

définir Température à température (° C)
si Température < 20
alors montrer l'icône

Annexe K

Fiche "élève" : Découvertes des entrées et sorties

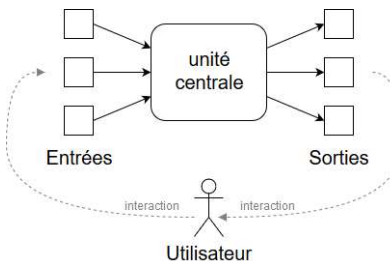
Découverte des Entrées et Sorties

Qu'est-ce qu'un ordinateur ?

Un *Ordinateur* est une machine capable de traiter beaucoup d'information et de réaliser de nombreux calculs à la suite. Ils sont présents un peu partout dans notre vie de tous les jours. Par exemple, les smartphones, les tablettes et les consoles sont aussi des ordinateurs.

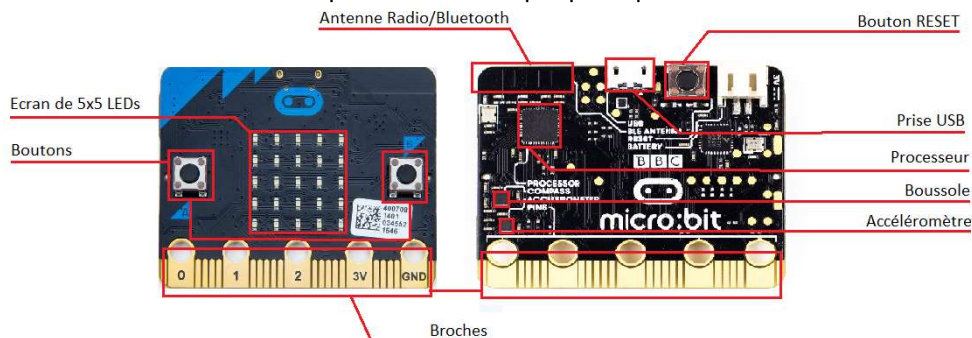
Deux grandes parties composent l'ordinateur :

- L'*Unité Centrale* est le cerveau de l'ordinateur. Elle va récupérer des données, les manipuler et les traiter.
- Les *Périphériques d'Entrée et de Sortie* permettent à l'utilisateur et à l'ordinateur d'interagir ensemble.
 - Les périphériques d'**Entrée** sont comme les cinq sens de l'ordinateur. Ça peut être un clavier, une souris, un micro, une manette de jeu, une webcam, etc.
 - Tandis qu'un écran, une imprimante ou encore des haut-parleurs sont des périphériques de **Sortie**.



Découvertes des entrées et sorties sur le micro:bit

Le micro:bit est un micro-ordinateur possédant divers périphériques d'entrées et sorties :



En plus de cette partie visible, le micro:bit comprend aussi des capteurs de luminosité (se trouvant sous les LEDs).

Avec ce petit appareil, on peut réaliser plein de choses. Par exemple : une alarme, des instruments électroniques, des jeux, une boussole, des petits robots, etc.

Pour aller plus loin

Si vous voulez en apprendre plus sur les ordinateurs, vous pouvez aller voir sur YouTube « C'est pas Sorcier – Ordinateur, tout un programme » (<https://www.youtube.com/watch?v=c96KP5jZVYk>)

Annexe L

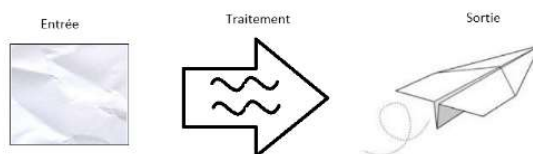
Fiche "élève" : Introduction à la programmation

Introduction à la Programmation

Un ordinateur est une machine capable de réaliser beaucoup de calculs compliqués très rapidement. Mais cette machine est aussi assez bête car il faut lui dire quoi faire. Pour que l'ordinateur fasse ce que l'on souhaite, il faut écrire un programme correspondant à une liste d'action que l'ordinateur doit exécuter.

Qu'est-ce que la programmation ?

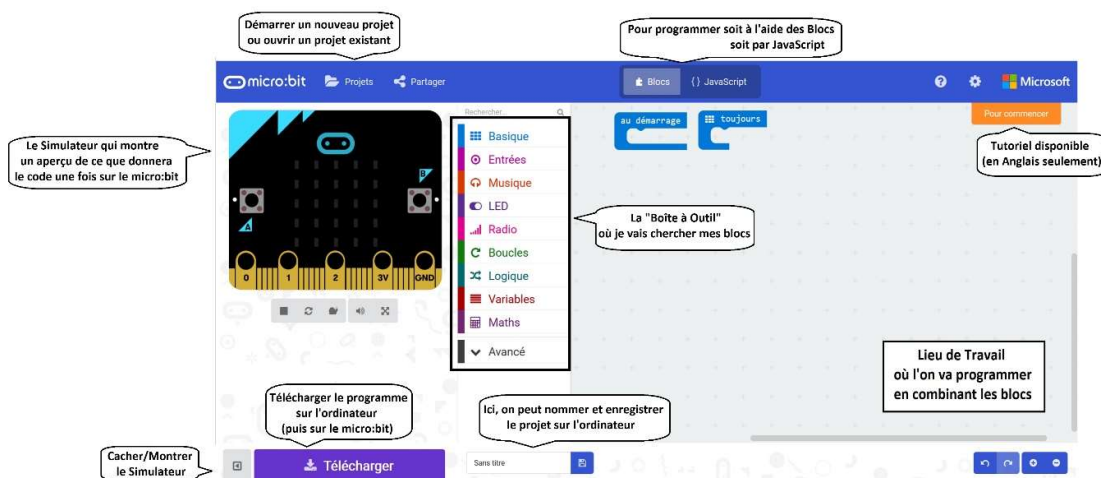
En exécutant ce programme, l'ordinateur va traiter des données selon un certain traitement (aussi appelé *Algorithme*, soit une suite d'instruction/d'action) afin de produire un résultat pour l'utilisateur. Par exemple, si j'ai une feuille de papier en entrée et que j'effectue avec une liste d'instructions de pliage, je pourrais obtenir en résultat un avion en papier.



Un autre exemple serait celui de la recette de cuisine. Si on a des ingrédients en entrée et que l'on souhaite avoir un gâteau au chocolat en résultat, il faut écrire une recette avec une liste d'action à effectuer pour obtenir un gâteau en traitant les ingrédients.

Et la programmation sur le micro:bit ?

Le micro:bit est un micro-ordinateur ayant à sa disposition différents périphériques d'entrée et sortie. Il peut exécuter un programme que l'on a fait sur le site officiel (<https://makecode.microbit.org/>) dont l'interface se présente ainsi :



Pour aller plus loin

Si vous souhaitez en savoir plus sur ce qu'est un algorithme et ses différentes caractéristiques, vous pouvez aller voir sur Youtube la vidéo « LET'S ALGO 1 - C'est quoi un algorithme ? » sur la chaîne de 'Flech & Computers' (<https://www.youtube.com/watch?v=l-5xj9jQQkk>)

Annexe M

Fiche "élève" : Les variables et l'accéléromètre

Les Variables et l'Accéléromètre

Qu'est-ce qu'une variable ?

Afin de pouvoir traiter beaucoup d'informations et de données, il est parfois nécessaire pour l'ordinateur d'enregistrer dans sa mémoire certaines données pour les réutiliser plus tard, au moment voulu.

Pour cela, on va utiliser des variables. Une variable, c'est comme une boîte dans la mémoire qui porte un nom et qui contient une valeur. On peut mettre une valeur dans une variable ou modifier son contenu mais on peut aussi récupérer une valeur qui se trouve dans une variable.

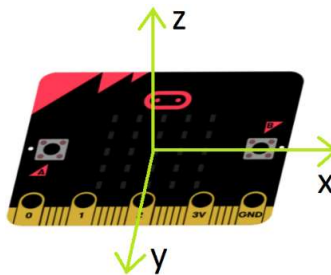
L'exemple suivant montre une suite d'instruction exploitant avec le concept de variable :

Étape	Instruction	Mémoire
1	a = 5;	a: 5
2	b = 2;	a: 5 b: 2
3	a = a + 3;	a: 8 b: 2
4	c = a * b;	a: 8 b: 2 c: 16
5	a = c;	a: 16 b: 2 c: 16
6	d = "Val";	a: 16 b: 2 c: 16 d: "Val"

On peut stocker dans une variable des nombres mais aussi des symboles comme des lettres ou même des messages entiers sous forme de suite de caractère.

Utiliser l'accéléromètre

Un accéléromètre est un capteur qui va mesurer l'accélération d'un objet. En d'autres termes, il va mesurer les mouvements de l'objet selon trois axes : x, y et z.



Et il va mesurer si l'objet est en mouvement, rapide ou lent, mais aussi dans quel sens on penche l'objet (vers la gauche, la droite, etc.). Ce type de capteur peut être utilisé de différentes manières comme pour les smartphones où l'accéléromètre permet d'orienter correctement l'affichage de l'écran (en portrait ou paysage) en fonction de l'orientation de l'appareil. La Wiimote et le Nunchuck de la Nintendo Wii utilise aussi des accéléromètres afin de mesurer directement les mouvements du joueur.

Pour aller plus loin

Si vous souhaitez en savoir plus sur les caractéristiques et l'affectation d'une variable, vous pouvez aller voir sur Youtube la vidéo « LET'S ALGO 5 - Les affectations » sur la chaîne de 'Flech & Computers' (<https://www.youtube.com/watch?v=71WinHtgc8A>)

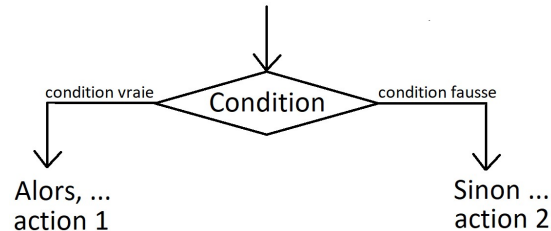
Annexe N

Fiche "élève" : Les conditionnelles et les capteurs

Les Conditionnelles et les Capteurs

Qu'est-ce qu'une conditionnelle ?

Une importante partie de la programmation est de pouvoir dire « Quand » un programme doit effectuer une certaine action, définir les conditions selon lesquelles doivent se réaliser une action.



Dans une instruction conditionnelle, il faut définir une condition qui a pour valeur 'vrai' ou 'faux'. Si cette condition est 'vraie', alors l'instruction va exécuter l'« action 1 » mais si la condition est 'fausse', alors l'instruction va exécuter l'« action 2 ». On n'exécutera soit l'« action 1 », soit l'« action 2 » en fonction de la condition.

Un exemple de conditionnel serait « SI il pleut ALORS reste à la maison SINON va à la plage » où on a deux actions différentes et l'on va effectuer l'une des deux actions en fonction de la condition.

Utiliser les capteurs

Afin que l'ordinateur puisse récupérer des données et des informations sur l'environnement qui nous entoure, il va souvent utiliser des capteurs. Ces capteurs vont transformer une grandeur physique en signal électrique. Cela est utile pour l'ordinateur pour effectuer certaines actions lorsqu'une condition est remplie. Par exemple, les capteurs suivants :

- La boussole : La boussole est un outil qui détecte les champs magnétiques comme le champ magnétique de la Terre et permet ainsi de repérer le Pôle Nord. Elle fonctionne donc moins bien entourer d'objets métalliques.
- Le thermomètre : afin de récupérer la valeur de la température. Cela est utile pour un thermostat qui va chauffer ou pas une pièce en fonction de la température ambiante.
- Les capteurs de luminosité : La valeur du niveau d'intensité lumineuse varie en fonction de la luminosité ambiante. Ils peuvent être utiles pour allumer automatiquement la lumière s'il commence à faire trop sombre.

D'autres capteurs sont utilisés au quotidien comme des capteurs de proximité sur les voitures pour indiquer au chauffeur s'il n'est pas trop proche d'un obstacle.

Annexe O

Fiche "aide" : La calculette

La Calculette

Réaliser une Calculette, suivez les étapes suivantes :

- 1) « Au Démarrage » du programme,
 - a. Initialiser une Variable à 0 (Nombre1 = 0 ;)
 - b. Initialiser une deuxième Variable à 0 (Nombre2 = 0 ;)
- 2) « Lorsqu'on appuie sur le bouton A »
 - a. Augmenter la valeur de la première Variable de 1
 - b. Montrer la valeur de la première Variable (Nombre1 = Nombre1 + 1 ;)
- 3) « Lorsqu'on appuie sur le bouton B »
 - a. Augmenter la valeur de la seconde Variable de 1
 - b. Montrer la valeur de la seconde Variable (Nombre2 = Nombre2 + 1 ;)
- 4) « Lorsqu'on incline à gauche »
 - a. Montrer la somme des valeurs des deux variables
- 5) « Lorsqu'on incline à droite »
 - a. Montrer la différence des valeurs des deux variables

Pour vous aider, vous pouvez trouver les blocs suivants dans la catégorie « Variable » :



: Ce bloc correspond à la variable « a »



: Ce bloc correspond à l'instruction « a = 0 ; »



: Ce bloc correspond à l'instruction « a = a + 1 ; »

Annexe P

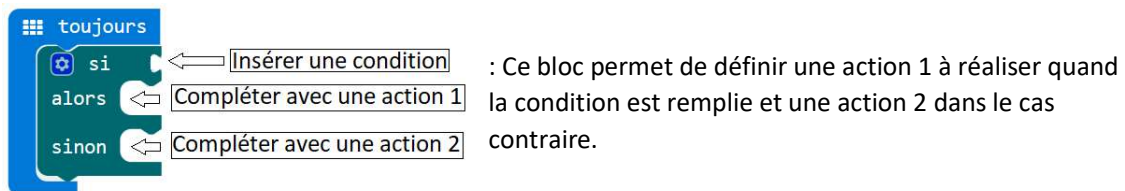
Fiche "aide" : Affichage Météo

L’Affichage Météo

Pour réaliser un Affichage Météo :

- 1) **SI** la luminosité est au-dessus de 40
 - a. **ALORS** montrer un Soleil !
 - b. **SINON**
 - i. **SI** la luminosité est au-dessus de 15
 1. **ALORS** montrer un Nuage !
 2. **SINON** Montrer une Lune!

Pour vous aider, vous pouvez trouver les blocs suivants dans la catégorie « Logique » :



Et le bloc suivant, trouvable dans la catégorie « Entrée », renvoie la valeur de la luminosité :



L’Affichage Météo ★★

Une fois terminé, vous pouvez améliorer votre Affichage Météo en rajoutant ces fonctionnalités :

- 1) « Lorsque j’appuie sur le bouton A »
 - a. **SI** la luminosité est en dessous de 40 **ET** la température est supérieure à 15°
 - i. **ALORS** montrer un parapluie
- 2) « Lorsque j’appuie sur le bouton B »
 - a. Montrer la température.

Pour vous aider, le bloc suivant renvoie la valeur de la température et est trouvable dans la catégorie « Entrée » :



Annexe Q

Contribution à Ludovia#CH 2018

Comprendre l'ordinateur à travers un système informatique tangible, le micro:bit

Nicolas Théate, Anne Smal, Benoit Frénay, Julie Henry
Université de Namur (Belgique)
PreCISE, Namur Digital Institute (NADI)
[\[benoit.frenay.julie.henry@unamur.be\]](mailto:benoit.frenay.julie.henry@unamur.be)

Résumé. La question posée ici est de savoir si un système tangible, le micro:bit en l'occurrence, constitue une aide à l'apprentissage par les enfants de ce qu'est un ordinateur, de comment il fonctionne et de quelles sont ses limites. Des éléments de réponse sont cherchés à travers la mise en place d'activités d'initiation à l'informatique et à la programmation auprès d'élèves âgés de 12 à 14 ans.

Mots-clés. Ordinateur, micro:bit, programmation, schéma fonctionnel, informatique

1 Introduction

Les technologies de l'information et de la communication (TIC) font partie intégrante de notre quotidien. Il est désormais presque impensable, notamment pour les jeunes, de s'en passer. On parle de génération Z, de *digital natives* (Baron & Bruillard, 2008)... Et si nous parlions plutôt de *digital natives*¹ ? Les usages intensifs rapportés² ne garantissent en rien la maîtrise des usages numériques. « *Surfer, lire des mails, utiliser les réseaux sociaux sont certes dans leurs pratiques courantes. (...) Par contre, traiter des données (structurer un texte, utiliser un tableur, un logiciel de messagerie), organiser l'information (la mémoriser, la faire circuler, la hiérarchiser...) (...) autant de domaines où l'expérience (...) montre que la plupart des jeunes se trouvent en difficulté. (...) La formation manque. Il ne suffit pas d'être « native numérique » pour maîtriser ces outils.* ». La Belgique ne fait pas figure d'exception en la matière, la formation au numérique et à ses usages étant presque totalement absente des cursus scolaires (Henry & Joris, 2016 ; 2015). Il est par ailleurs étonnant de constater que certains jeunes répondent « non » à la question « votre smartphone est-il un ordinateur ? ». Plus qu'un problème lié à l'évolution du vocabulaire, il ressort que peu s'interrogent sur ce qu'est réellement un ordinateur, sur son fonctionnement, sur ses capacités (et ses limites) et sur les moyens existants d'interagir avec lui. Or il apparaît désormais important de comprendre les systèmes qui régissent l'usage des outils numériques pour devenir acteur (et non seulement consommateur) d'une société de plus en plus numérique. « *Il importe peu de comprendre les détails du fonctionnement très complexe d'un processeur ou d'une carte graphique. Il est par contre essentiel de maîtriser les bases de l'algorithmique et de sa mécanique du raisonnement... et pour des questions de performance, il peut être utile de comprendre où l'information que nous utilisons est stockée, en mémoire, sur disque ou sur le réseau. Surtout, il est indispensable de comprendre le sens de cette information, comment elle est représentée, comment elle est organisée...* » (Abiteboul³).

Des activités à destination d'élèves issus du secondaire inférieur (12 à 15 ans) ont été développées avec comme objectif de leur enseigner le principe de fonctionnement d'un ordinateur (schéma fonctionnel). L'originalité de ces activités tient dans le choix d'utiliser un système informatique tangible, le micro:bit⁴. Il s'agit de déterminer en quoi ce type de système peut s'avérer utile dans la construction des représentations des élèves concernant l'ordinateur.

En première partie, le contexte et les objectifs de recherche seront présentés. La méthodologie sera décrite brièvement en partie 2. La partie 3 consiste en une première discussion des résultats obtenus à l'heure d'écrire cet article. Enfin, une courte conclusion clôturera cet apport.

1 http://next.liberation.fr/vous/2010/03/10/les-jeunes-ne-sont-plus-interesses-par-l-outil-ordi_614226, consultée en ligne le 23 février 2018

2 Les jeunes de 13 à 17 ans passent en moyenne cinq heures par jour en ligne – <http://www.eureasso.fr/dossier-les-jeunes-et-l-engagement>, consultée en ligne le 23 février 2018

3 Leçon inaugurale du Collège de France prononcée le jeudi 8 mars 2012. Chaire d'Informatique et Sciences Numériques - <http://books.openedition.org/cdf/529#ftn1>, consultée en ligne le 23 février 2018

4 <https://www.microbit.co.uk/>, consultée en ligne le 12 décembre 2017

2 Contexte, ancrages théoriques et objectifs

Dans le contexte d'enseignement posé notamment par Abiteboul, à savoir l'introduction à tout un chacun des concepts de base de l'informatique, nous nous sommes intéressés au système informatique tangible développé par la BBC et distribué massivement de par le monde, le micro:bit (cfr. Figure 1). Ce micro-ordinateur possède et rend « visible », contrairement à certains systèmes « robots » (thymio⁵, Dash⁶, etc.), tous les composants nécessaires pour arriver à nos fins : un processeur, une mémoire, divers périphériques d'entrée et de sortie tels que des boutons, des capteurs (thermomètre, boussole, accéléromètre, entre autres) et un écran de LEDs. Il possède également des broches permettant de brancher des périphériques additionnels (buzzer ou servomoteur, par exemples) et étendre ainsi ses capacités. Le micro:bit permet l'exécution d'un programme écrit en langages Javascript ou Python, ou encore à l'aide d'interfaces web de programmation par blocs⁷. En ce qui concerne l'enseignement de la programmation (dans un contexte d'une introduction auprès d'enfants), l'intérêt de langages visuels en blocs est régulièrement souligné : « *it can be used to provide young people with a positive introduction to computing* » (Adams, 2010, p.360), « *visual programming languages, like Scratch, offer the experience of syntax free programming which is suitable for novices.* » (Mladenović et al., 2017, p.7). L'intérêt du micro:bit est lui grandissant : « *the results from this research assert that pupils believe that the BBC micro:bit is an enjoyable, easy to use device that is beneficial when learning how to code, with various other skills being developed in tandem* » (Gibson & Bradley, 2017, p. 36). De notre point de vue, le choix de cet outil se justifie, entre autres, par le souhait de mettre à la disposition des enfants un système qui leur permette de faire facilement le lien entre les différents composants de ce dernier et le côté abstrait de la programmation : « *Even something as simple as projecting a display on top of paper, or on a tabletop with physical objects, could be used as an effective technique for getting children to see the relationships between objects and representations they create and manipulate in the physical world, and what happens in the computational world.* » (O'Malley, 2004, p.34). Dès lors, il s'agit ici de déterminer comment exploiter au mieux le micro:bit (et de façon générale, les systèmes tangibles apparentés) pour faire comprendre aux jeunes ce qu'est un ordinateur ? Comment leur faire prendre conscience des capacités et des limites de ce dernier ? Des éléments de réponse à ces questions seront à tirer du développement et de la mise en place d'activités d'initiation à l'informatique auprès d'un public constitués d'enfants de 12 à 15 ans.

3 Méthodologie

La première phase de cette recherche consiste en la création d'activités à destination des enfants. Pour ce faire et pour assurer leur acceptation par les enseignants - pour la majorité non formés (Henry & Smal, 2018 ; Henry & Joris, 2013) -, un travail étroit est réalisé à la fois avec des pédagogues et avec les enseignants eux-mêmes. Cinq établissements scolaires ont été désignés « école-pilote » et ont mis à notre disposition leurs élèves inscrits dans les deux premières années de l'enseignement secondaire. De ce fait, toute activité créée se voit validée en contexte réel, au sein d'une classe, et modifiée, le cas échéant, selon une démarche itérative.

En ce qui concerne la compréhension qu'ont les jeunes de l'ordinateur et la mesure d'une probable modification de celle-ci, les élèves sont soumis à des pré et post-tests encadrant une séquence composée d'une dizaine d'heures d'activité (et présentée en section 4). Ces tests, identiques pour faciliter la comparaison et mesurer une évolution, consiste en deux questions ouvertes demandant à l'élève de définir l'ordinateur et de décrire ce que cet ordinateur est capable/incapable de faire. Les résultats obtenus ont fait l'objet d'une analyse multidimensionnelle au moyen du logiciel IraMuTeQ⁸.

4 Résultats et discussion

Quelle représentation de l'ordinateur transmettre aux enfants ? Le choix de l'auteur s'est porté sur celle d'une machine (électronique) capable d'exécuter une suite d'actions pour lesquelles elle a été programmée. Cette machine peut être décomposée en trois parties, clairement identifiables sur son schéma fonctionnel (cfr. Figure

5 <https://www.thymio.org/fr/thymio>, consultée en ligne le 26 février 2018

6 <https://www.makewonder.com/dash>, consultée en ligne le 26 février 2018

7 <https://www.microbit.co.uk/create-code> ; <https://microbit.edublocks.org/>, consultées en ligne le 23 février 2018

8 <http://www.iramuteq.org/>, consultée en ligne le 14 décembre 2017

2) : l'unité centrale, les périphériques d'entrée et ceux de sortie. L'unité centrale comprend le processeur et la mémoire. Les périphériques d'entrée et de sortie rendent possibles les interactions entre la machine et l'utilisateur.

Figure 1. Le micro:bit

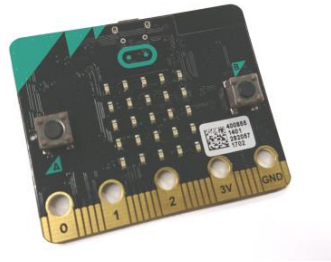
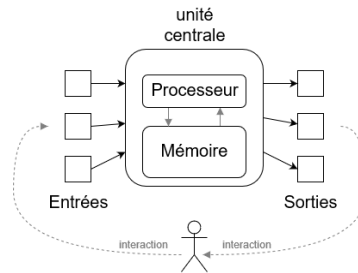


Figure 2. Schéma fonctionnel de l'ordinateur



Quatre activités (cfr. Tableau 1) ont été développées pour transmettre au mieux cette représentation simplifiée de l'ordinateur. Celles-ci mettent notamment en jeu les différents périphériques du micro:bit, à savoir notamment ses capteurs.

Tableau 1. Activités composant la séquence d'initiation à la programmation

Intitulé	Description	Objectifs
Premier pas avec le micro:bit – Découverte des entrées et sorties	Découvrir les capacités du micro:bit à partir de programmes préinstallés et identifier les entrées et sorties	Apprendre à manipuler le micro:bit et savoir identifier ses composants
Premier pas avec le micro:bit – Introduction à la programmation	Découvrir l'interface web de programmation par blocs associée au micro:bit	Savoir programmer à l'aide de l'interface web et transférer un programme sur le micro:bit
Variable et accéléromètre	Réaliser une calculatrice utilisant variables et capteur (ici, l'accéléromètre)	Comprendre et savoir utiliser les variables et l'accéléromètre
Conditionnelle et capteurs	Réaliser à l'aide de structures conditionnelles un affichage de la météo utilisant les capteurs de luminosité et le thermomètre	Comprendre et savoir utiliser les structures conditionnelles et les capteurs

La première activité consiste, dans un premier temps, en une présentation du schéma fonctionnel de l'ordinateur. L'objectif visé est de faire prendre conscience aux élèves que ce type de schéma est générique aux systèmes informatiques, y compris le micro:bit. Ce dernier n'est pas considéré comme un ordinateur par la majorité d'entre eux. L'identification, en seconde partie d'activité, des différents périphériques disponibles via la manipulation de micro:bit pré-programmés et le « placement » de ces périphériques dans le schéma fonctionnel permet pourtant aux enfants de rapidement faire le parallélisme entre le micro:bit et un ordinateur.

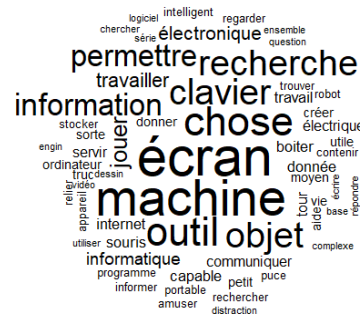
La seconde activité consiste en une initiation à la programmation à travers la découverte de l'interface « officielle » de programmation en blocs associée au micro:bit. Sans que soient encore explicités les différents concepts de base, les élèves sont laissés en situation de création, se contentant d'emboîter les blocs dits de base (affichage, principalement). Ils s'essaient alors au transfert de leur premier programme sur le système tangible dans le but de pouvoir le tester.

Viennent ensuite deux activités liées aux concepts de base de programmation que sont les variables et les structures conditionnelles. L'activité sur les variables propose de programmer une calculatrice (utilisant l'accéléromètre). Les élèves manipulent deux variables et effectuent des opérations mathématiques dépendant de l'orientation de l'appareil. L'activité sur les structures conditionnelles met en jeu le capteur de luminosité pour faire prendre conscience aux enfants que certains comportements d'un programme (et donc d'un ordinateur) peuvent dépendre de l'environnement, d'événements et/ou de contextes précis. Les élèves vont alors programmer le micro:bit pour qu'il affiche un symbole différent (soleil, lune ou nuage) en fonction de la luminosité ambiante.

À l'heure d'écrire cet article, les deux premières activités ont pu être testées auprès de plus de 200 élèves âgés de 12 à 14 ans, présentant des profils très variés. Une analyse complète des pré-test est en cours. Les post-tests sont n'ont pas encore tous été collectés. Concernant les pré-tests, certains constats ont déjà pu être faits. Une première analyse partielle à l'aide du logiciel IraMuTeQ des résultats obtenus pour la question « Pour toi, un ordinateur, c'est... » montre que les enfants pensent avant tout à une machine/un outil auxquels sont connectés un clavier et un écran. Il apparaît également clairement que l'ordinateur est utilisé surtout pour rechercher de l'information,

pour jouer et pour travailler (cfr. Figure 3). Les résultats des post-tests devraient permettre de mettre en évidence l'apport d'un système tangible (et de l'accessibilité facilitée de ses composants) dans la représentation qu'ont les enfants de l'ordinateur, ici fortement liée à l'aspect physique d'un ordinateur fixe/portable tel qu'il est rencontré dans la majorité des familles.

Figure 3. L'ordinateur vu par les enfants



L'interface web de programmation s'est avérée très intuitive pour les jeunes. Des difficultés ont cependant été ressenties, notamment par rapport à la mauvaise traduction (française) de certains blocs ayant comme conséquence de créer une confusion chez les élèves. Ceux-ci ont également été perturbés par la quantité trop importante de blocs mis directement à leur disposition. Ces quelques observations fortuites appuient la décision prise de découvrir l'interface de façon « guidée », pas à pas, en expliquant certains blocs : comme l'incrémentation de variable, par exemple.

5 Conclusion

Afin de retravailler la représentation qu'ont les jeunes d'un ordinateur et les aider ainsi à mieux le comprendre (et par extension, à mieux comprendre la technologie), le micro:bit s'avère être un outil prometteur pour mettre en place des activités d'introduction aux ordinateurs et à la programmation. De par sa caractéristique tangible, le micro:bit permet de s'aider d'un support physique pour mieux appréhender des concepts tels que le schéma fonctionnel d'un ordinateur, voire les concepts de base en programmation. À partir de tests passés auprès de plus de 200 élèves âgés de 12 à 14 ans, une analyse va être réalisée pour mesurer l'impact d'un système tangible sur la représentation (et la compréhension) qu'ont les jeunes de l'ordinateur. Les premiers résultats sont présentés ici brièvement, pour poser le décor.

Références

- Adams, J.C. (2010) *Scratching Middle Schoolers's Creative Itch*. ACM, 356-360
- Baron, G.-L. & Bruillard, E. (2008). Technologies de l'information et de la communication et indigènes numériques : quelle situation ? *Rubrique de la revue STICEF*, Volume 15.
- Gibson, S. & Bradley, P. (2017). A study of Northern Ireland key stage 2 pupils' perceptions of using the BBC micro:bit in STEM education. *The STeP Journal Student Teacher Perspectives*, Vol 4 (1), 15-41.
- Henry, J., & Joris, N. (2015). Le bagage TIC des étudiants en Belgique francophone. État des lieux. *Informatique en éducation: perspectives curriculaires et didactiques*, 61-81.
- Henry, J., & Joris, N. (2016). Informatics at secondary schools in the French-speaking region of Belgium: myth or reality?. *ISSEP 2016*, 13.
- Mladenovic, M., Krpan, D. & Mladenovic, S. (2017). *Learning programming from Scratch*. International Conference on New Horizons in Education 2017.
- O'Malley, C., & Stanton Fraser, D. (2004). *Literature review in learning with tangible technologies*. Bristol: Futurelab.

Annexe R

Proposition d'atelier pour Ludovia#CH 2018

Introduction à l'ordinateur et à l'informatique à l'aide d'un système informatique tangible, le micro:bit

Niveau scolaire concerné : Premier degré de l'enseignement secondaire inférieur (12-14 ans)

Mots Clés : Découverte / Programmation / Ordinateur / Education numérique / Capteur / Objets Connectés / Technologie / micro:bit / Système tangible

Problématique pédagogiques : En plusieurs années, la technologie est devenue omniprésente dans le quotidien. Ordinateurs, smartphones, objets connectés, etc. il est impensable aujourd'hui pour un jeune de se passer de la technologie tant elle est présente et facilite la vie de tous. Et pourtant, leurs visions de la technologie et des ordinateurs restent floues.

Aujourd'hui, avec la démocratisation des objets connectés, les télévisions, les voitures ou encore certains outils ménagers sont devenus de vrais ordinateurs.

Des activités sont donc développées pour permettre aux jeunes de démystifier l'ordinateur en leur introduisant les concepts de base d'un ordinateur (comme le schéma fonctionnel avec processeur, mémoire et périphériques d'entrée et sortie) et ceux de la programmation, et cela à l'aide d'un système informatique tangible.

Apport du numérique ou présentation de la techno utilisée : Pour aider les jeunes à comprendre ce qu'est plus précisément un ordinateur et comment on interagit avec lui, le micro:bit est un micro-ordinateur composé de plusieurs capteurs. L'appareil dispose de boutons, de broches pour créer des circuits électriques, d'un écran de 25 LEDs et de capteurs comme un accéléromètre pour mesurer les mouvements, un thermomètre, des capteurs de luminosité ou encore une boussole. La plupart des composants essentiels du micro:bit, comme son processeur et certains capteurs, sont visibles à l'œil nu. Le micro:bit est purement programmable et est capable d'exécuter un programme écrit en langage Javascript ou Python, ou encore à l'aide d'une interface web de programmation par blocs (<https://www.microbit.co.uk/create-code>). Il permet donc de faire le lien entre l'aspect abstrait de la programmation et un aspect plus concret de l'appareil physique lui-même.

Dans l'interface proposée pour le micro:bit comme dans beaucoup d'interfaces similaires (Blockly, Scratch, etc.), les blocs correspondent à des instructions permettant d'exploiter notamment les diverses fonctionnalités du système tangible. Facilement combinables entre eux, à l'instar des pièces de puzzle, les différents blocs représentent les règles syntaxiques d'un programme, comme des règles d'emboîtement de pièces, guidant l'utilisateur.

Relation avec le thème de l'édition : Apprendre ce qu'est un ordinateur et comment programmer peut permettre à l'enfant de s'émanciper dans le monde qui l'entoure, lui permettre de mieux comprendre la place de la technologie dans son environnement et comment on interagit avec elle.

Il apparaît désormais important de comprendre la technologie, plus que d'en être seulement un consommateur, pour devenir le citoyen éclairé d'un monde de plus en plus numérique.

Synthèse et apport du retour d'usage en classe : Avec ce système informatique tangible qu'est le micro:bit, il est possible de proposer des activités en classe pour permettre de retravailler la représentation qu'on les jeunes d'un ordinateur. Au cours de ces activités, les jeunes élèves élargiront

leur vision de l'informatique et découvriront des concepts basiques de l'informatique comme l'algorithmique, les variables ou les structures conditionnelles, utiles pour mieux comprendre le fonctionnement des technologies qui nous entourent. Ces activités ont été validées dans différentes écoles secondaires et ont permis de récolter différentes données au moyen de tests à compléter avant et après les activités. Ces dernières sont en cours d'analyse mais les premiers résultats montrent déjà que leur vision de l'ordinateur est souvent réduite à celle d'une machine avec un écran et à certaines de ses fonctionnalités comme pouvoir communiquer ou effectuer des recherches. Il est donc intéressant de pouvoir élargir cette vision et celle de l'informatique en général.

